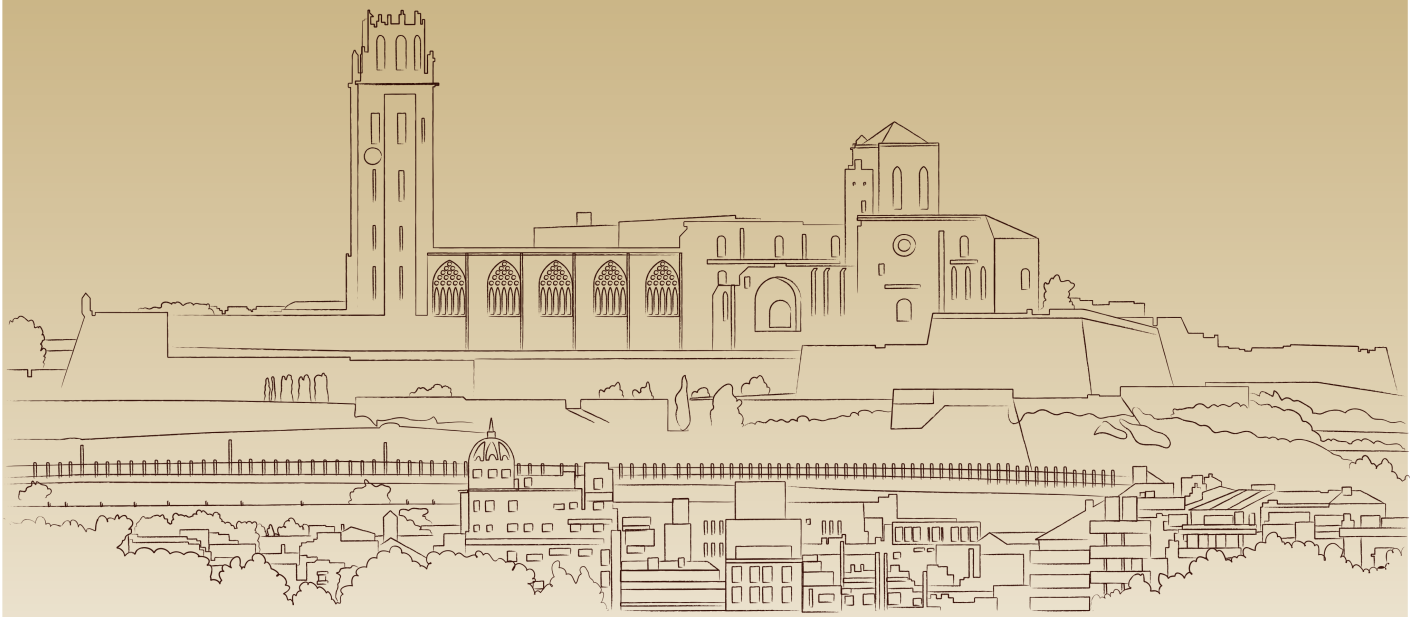
14è Congrés Internacional de l'Associació Catalana d'Intel·ligència Artificial

Lleida, 26, 27 i 28 d'octubre de 2011



http://ccia2011.udl.cat

Organitzadors: Artificial Intelligence Research Group — acia

Col·laboradors: Universitat de Lleida — Universitat Pompeu Fabra — Centre de Computació de Ponent — Diputació de Lleida — IIIA Institut d'Investigació en Intel·ligència Artificial — turóseuvella

Preface

This year, 2011, is very special for the national research community working on Artificial Intelligence (AI): the International Joint Conference on Artificial Intelligence (IJCAI) was held in Barcelona, and the Catalan Association for Artificial Intelligence (ACIA) played a key role in the organization of the event. With more than 1400 participants, the IJCAI in Barcelona has been indeed the most attended IJCAI of the last 20 years.

Despite the significance and symbolism of IJCAI, the Catalan AI community gathered also for its annual conference, which this year reaches its 14th edition. The 14th International Conference of the Catalan Association for Artificial Intelligence, CCIA-2011, has been held at the Universitat de Lleida, from October 26th to October 28th, and has been organized by the AI groups of Universitat de Lleida (UdL) and Universitat Pompeu Fabra (UPF), and the Artificial Intelligence Research Institute (IIIA-CSIC).

CCIA is a conference intended to bring together researchers from different areas of the Catalan-speaking AI community, and promote the cooperation among the local research groups. Topics of interest include, but are not limited to agents and multi-agents systems, constraints and satisfiability, evolutionary computing, knowledge representation, machine learning, natural language, planning, reasoning models, robotics, search, and uncertainty.

This book contains the papers accepted for presentation at CCIA-2011. All the received submissions were rigorously reviewed by the Program Committee, and each paper received three independent reviews. Out of 28 submissions, 24 papers were accepted for inclusion in the book and for oral presentation at the conference. The conference program also featured two invited talks by two outstanding international researchers: Christian Bessière (CNRS, Université de Montpellier, France), and Bart Selman (Cornell University, USA).

We would like to express our sincere gratitude to the authors of the contributed papers, to the invited speakers for their enlightening talks, to the Program Committee and the additional reviewers for their careful and thorough work, to the Publicity Chair for promoting the conference and developing the webpage, and to the Organizing Committee for taking care of the organization of the event itself.

Our special thanks go also to the president of ACIA, Vicenç Torra, who asked us to serve as Local Arrangement Chair, Program Chair, and General Chair of CCIA-2011.

Financial and organizational support was generously provided by Diputació de Lleida, Universitat de Lleida (UdL), Universitat Pompeu Fabra (UPF), and the Artificial Intelligence Research Institute (IIIA-CSIC).

We hope that the present book gives a glimpse of current AI research in Catalonia, and inspires new ideas and further work.

Lleida, October 2011

Cèsar Fernández (UdL), Local Arrangement Chair
Hector Geffner (ICREA & UPF), Program Chair
Felip Manyà (IIIA-CSIC), General Chair

Organization

Conference Organization

General Chair	Felip Manyà, IIIA-CSIC
Programme Chair	Héctor Geffner, ICREA & UPF
Local Arrangement Chair	Cèsar Fernández, UdL
Publicity Chair	Carles Mateu, UdL

Program Committee

Núria Agell, ESADE
René Alquézar, UPC
Teresa Alsinet, UdL
Cecilio Angulo, UPC
Carlos Ansótegui, UdL
Josep Argelich, UdL
Federico Barber, UPV
María Luisa Bonet, UPC
Ramón Béjar, UdL
Miquel Bofill, UdG
Dídac Busquets, UdG
Hubie Chen, UPF
Ulises Cortés, UPC
Pompeu Casanovas, UAB
Pilar Dellunde, IIIA-CSIC
Marc Esteva, IIIA-CSIC
Ana García Fornés, UPV
Elisabet Golobardes, URL
José Manuel Iñesta, UA
Anders Jonsson, UPF
Vicente Julián, UPV
Javier Larrosa, UPC
Jordi Levy, IIIA-CSIC
Ramon López De Mántaras, IIIA-CSIC
Maite López Sánchez, UB
Beatriz López, UdG
Joaquim Meléndez, UdG
Pedro Meseguer, IIIA-CSIC
Antonio Moreno, URV
Angela Nebot, UPC
Eva Onaindia, UPV
José Oncina, UA
Santi Ontañón, IIIA-CSIC
Jordi Planes, UdL
Josep Puyol Gruart, IIIA-CSIC
Miguel Ramírez, UPF
David Riaño, URV
Jordi Sabater Mir, IIIA-CSIC
José Salvador Sánchez Garreta, UJI
Miquel Sànchez Marrè, UPC
José Miguel Sanchiz, UJI
Ricardo Toledo, UAB
Vicenç Torra, IIIA-CSIC
Carme Torras, IRI-CSIC/UPC
Aïda Valls, URV
Maria Vanrell, UAB
Xavier Varona, UIB
Juan Miguel Vilar, UJI
Mateu Villaret, UdG
Jordi Vitrià, UB

Additional Reviewers

Alba Cabiscol
Francesc Moreno-Noguer
Arnau Ramisa
Victor Sánchez
Andrea Schumm
José Such
Johan Thapper

Sponsoring Institutions

Associació Catalana d'Intel·ligència Artificial (ACIA)
Artificial Intelligence Research Group (UdL)
Centre de Computació de Ponent (CCP-UdL)
Diputació de Lleida
Institut d'Investigació en Intel·ligencia Artificial (IIIA-CSIC)
Universitat de Lleida (UdL)
Universitat Pompeu Fabra (UPF)

Contents

An Assistance Infrastructure for open MAS[1]

Pablo ALMAJANO [a,b], Maite LÓPEZ-SÁNCHEZ [b], Marc ESTEVA [a] and Inmaculada RODRÍGUEZ [b]

[a] *Artificial Intelligence Research Institute (IIIA) CSIC, Barcelona, Spain*
[b] *Universitat de Barcelona, Barcelona, Spain*

Abstract. Organisations are an effective mechanism to define the coordination model that structure agent interactions in Open MAS. Execution infrastructures mediate agents interactions while enforcing the rules imposed by the organisation. Although infrastructures usually provide open specifications to agents, understanding this specification and participating in the organisation could result a difficult task to agents, specially when the system is hybrid (i.e participants can be both human and software agents) and its specification becomes more and more complex. In this paper, we formalize an Assistance Infrastructure for Hybrid open MAS that helps agents to pursue their particular goals and, when they are aligned with global goals, lead to a better system's global performance. With this aim, we propose four advanced services (offered by the assistance infrastructure in a distributed way): (1) refined information, (2) justification, (3) advice and (4) estimation. We define two types of assistant agents. A Personal Assistant provides direct and sole support to one agent while a Group Assistant performs those complex processes which affect a group of participants with common services of interest.

Keywords. coordination support, organisations and institutions, self-organisation, agents assistance

Introduction

Usually, multi-agent systems (MAS [15]) design and implementation involves the specification of a coordination model and the development of the infrastructure in charge of enacting it. In open MAS, systems are populated by heterogeneous agents trying to achieve particular and/or collective goals. These agents are developed by third parties so the number and kind of agents that may participate in an open MAS is unknown at development time, and varies at runtime [16]. Organisations have proven to be an effective mechanism to define the coordination model that structures agent interactions in open MAS, and infrastructures give support to their execution imposing the rules established by the organisation. In Organisation Centred MAS (OCMAS [12]) approaches, these infrastructures define frameworks where agents with different cognitive abilities may interact. Although OCMAS infrastructures usually provide open specifications to agents [10] [14], understanding this specification and participating in the organisation could result a difficult task to agents, specially as its specification becomes more and more complex.

[1]This work is partially funded by EVE (TIN2009-14702-C02-01 / TIN2009-14702-C02-02) and AT (CONSOLIDER CSD2007-0022) Spanish research projects, EU-FEDER funds and the Catalan government (Grant 2005-SGR-00093). Marc Esteva enjoys a Ramón y Cajal contract from the Spanish Government.

If we take the humans in the loop and consider hybrid systems, where agents may be humans or software agents, the complexity increases and facilitating agent participation becomes a mandatory issue [2] [19].

This paper focuses on the challenge of improving agents' participation in the organisation by means of an *Assistance Infrastructure*. Certain knowledge about the organisation and its environment require complex computational processes in order to be useful for agents (e.g. planning or estimation). Therefore, agents would improve their participation in the organisation if the infrastructure could provide them with some assistance mechanisms that facilitate such processes. Our aim is to help agents in achieving their goals, and, when they are aligned with global goals, lead to a better system's global performance [18].

In this paper we further formalise the set of Agent's *Assistance Services* defined by Campos et al. [6]. We propose *Assistance Services* to offer, in addition to basic organisational information, more complex services such as (1) elaborated *Information*, as an example, specific statistics; (2) *Justification* of the consequences of a performed action; (3) giving *Advice*, for instance providing a plan to achieve agent's goals; and (4) *Estimation* of the consequences of performing an agent's action prior to its execution. The *Assistance Infrastructure* is the framework extension that enables the proposed *Assistance Services* in an open OCMAS. This extension is added to the system as an additional component we call the *Assistance Layer*. Inside this layer, two assistance agents offer the *Assistance Services*: i) *Personal Assistants* (*PA*s) and ii) *Group Assistants* (*GA*s). Each *PA* provides direct and sole support to one agent in the organisation. Nevertheless, we propose to group agents based on their properties regarding their services of interest (*Assisted Groups*) and provide the corresponding services to these groups. These services can be provided by a new level of assistants, that we denote as *GA*s. One *GA* provides support to all the *PA*s within its *Assisted Group* and therefore *PA*s computing time and communication overload may be alleviated.

The paper is structured in the following parts. First, section 1 summarizes the related work. Second, section 2 depicts amWater, our motivation example. Next, section 3 provides basic definitions and notation in order to formalise the proposed assistance services. Section 4 describes and formalises the *Assistance Infrastructure*. Finally section 5 gives some conclusions and future work.

1. Related work

There are two main lines of active research in assistance to MAS provided by *Software Agents*: organisational assistance services [7] [4] [5], and agent assistance services [9] [17] [8]. Regarding organisational assistance, Centeno et al. [7] defined an incentive mechanism (*Incentivators*) which induces individual participants to follow organisational goals by learning their preferences and doing modifications in the environment. On the other hand, Bou et al. [3] defined an Electronic Institution with autonomic capabilities that allows it to adapt its regulations to comply with institutional goals despite varying agent's behaviours (Autonomic Electronic Institutions). In a preceding work, they also applied Case-Based Reasoning (CBR) to reason about the process of adapting the norms of an Electronic Institution when certain system-wide measures differ from the expected ones [4]. Finally, the Two Level Assisted MAS Architecture (2-LAMA [5]) also provides organisational assistance services. It is composed by two levels. In the domain-level (DL) agents perform domain specific activities. On top of it, a distributed meta-level (ML) is in charge of providing assistance to the DL. This assistance is performed by changing

the norms of the organisation and it is provided to groups of not overlapped and fixed clusters of agents. Our approach goes in the line of agent assistance service, so it differs from previous ones in organisational perspectives. Moreover, our concept of groups follows the proposal by Ferber et al. [11] where groups are agent aggregations and one agent may be member of more than one group at time so that groups can freely overlap.

Other works focus on agent assistance services. Electric Elves [9] is a system that applies agent technology in service of the day-to-day activities of the Intelligent Systems Division of the University of Southern California Information Sciences Institute. Chalupsky et al. developed specific Software Personal Assistants (*SPA*) for project activities coordination and external meetings organisation. Since our proposal is general for MAS our *Assistance Layer* can include such kind of services.

These and other proposed *SPA*'s abilities were evaluated in a conceptual framework that simulated human behaviour in different MAS structures [18]. In this research, Okamoto et al. evaluated the impact that *SPA*s have on the individual performance and on the collective performance of the organisation as a whole. They built a computational model of human organisations and analysed two types of agent's organisational structures: hierarchical and horizontal. One *SPA* measured ability that is close to our proposal is the decision support (see section 4.1.4). They concluded that supporting decision tasks in human organisations increases the success rate (i. e., to meet the deadline with higher probability) and the speed performance average (i. e., to meet the deadline more rapidly), this is particularly the case in organisations with hierarchical structure.

A recent work presented a generic assistant agent framework in which various applications can be built [17]. As a proof of concept application, it implemented a coalition planning assistant agent in a peacekeeping problem domain. A more general framework for organising MAS [8] contains *Informative Organisational Mechanisms* and *Regulative Organisational Mechanisms*, a generalisation of the *Incentivators* [7]. As mentioned approaches, we also propose a general framework. Moreover, we propose to offer planning in our infrastructure (see section 4.1.3) as in the former. The latter, *Informative Organisational Mechanisms*, is a generalisation of our *Agent's Assistance Services*.

Existing OCMAS infrastructures already offer some kind of information about the organisation to a participant. In the $\mathcal{S} - \mathcal{M}oise^{+}$ [14] middleware, the OrgManager provides useful information for the organisational reasoning of agents and its organisational coordination. In this model, an agent is allowed to know another agent information if their roles are linked by an acquaintance relation (defined in the social level). Moreover, the OrgManager also informs actors about the new permissions, obligations, and goals they can pursue when a new state is reached in the organisation. Our framework provides similar information services and also more elaborated ones.

2. amWater: An Example of Assistance Scenario

Along our formal proposal of the assistance infrastructure we will use as example an electronic market of water rights (*amWater*). *amWater* is a simplification of *mWater* [13] which is an Electronic Institution (EI [1]) focusing on a water market extended with conflict resolution features. EI engine guarantees the correct execution of the system respecting the institutional rules and storing system state. *amWater* implements an electronic market associated to a specific basin. Many water basins are divided in geographical areas of interconnected water rights, i. e. water transfers between them are possible by using available basin's infrastructures. From now on, we will only refer to this type of water basins.

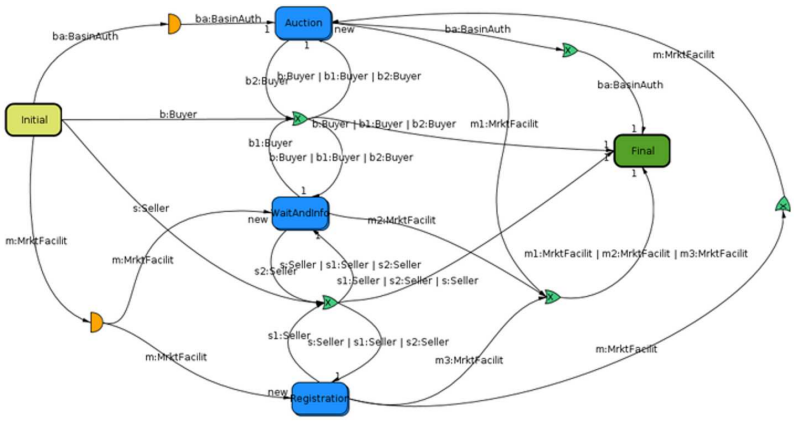


Figure 1. amWater Performative Structure

In our scenario, agents may adopt different roles. Irrigator agents may behave as either buyer or seller subroles while market facilitator and basin authority are staff agents. Figure 1 shows the performative structure [1] of amWater (i. e. the work-flow along several agent activities). It has three scenes which enact the market: *Registration* scene, where the market facilitator is in charge of registering sellers' rights; *Waiting and Information* scene, where irrigators can ask for information about auctions to the market facilitator; and *Auction* scene. The *Auction* scene is where the auction of water rights takes place. We have selected the Japanese Auction protocol because its characteristics seem appropriate for the evaluation of the *Assistance Layer*. There are three roles involved in this scene: buyers (bidders), the market facilitator (auctioneer) and the basin authority announces the valid agreements. The organisation creates one Auction scene for each area of the basin. Water rights are auctioned in consecutive rounds. Only buyers with owned water rights belonging to the area of the auction are allowed to join it.

3. Basic Definitions

To refer to the *organisation*, we consider the following definition of an organisation model from Campos et al. [5]: $Org = \langle SocStr, SocConv, Goals \rangle$. Its main concepts are: a social structure, its social conventions and some organisational goals. We consider open OCMAS and thus we assume agents (Ag) to not belong to the *organisation* itself.

3.1. Organisational Trace

An organisation evolves at run time with the interactions of the agents. As the result of these interactions, both organisation and agent's properties could experiment changes, modifying the organisation's state. In this way, the organisation goes through different execution states at run time. The circumstances that made the organisation evolve may be processed by our proposed *Assistance Layer* in order to offer information about past events (e. g., past auction results or statistics). Similarly, the *Assistance Layer* may analyse agents behaviour in order to help them to interact in the organisation (e. g., it could provide plans on what to do in a particular situation to achieve a particular goal).

Therefore, the *Assistance Layer* may provide support taking into account current and past execution states of the organisation. For this purpose, the *Assistance Layer* keeps the different execution states of the organisation (S). The information saved on S has to

be complete enough to characterize the state of the organisation and its agents. Moreover, for each step s of the simulation, S only contains non-static information of the organisation. We first define S_s as a tuple:

$$S_s = \langle SocStr_s, SocConv_s, Goals_s, AgP_s, EnvP_s \rangle \quad (1)$$

where:

– $SocStr_s$ are the identifiers of participating agents and their roles at step s.

– $SocConv_s$ are the social conventions. It is composed by $Prot_s$ and $Norms_s$: $SocConv_s = \langle Prot_s, Norms_s \rangle$. $Prot_s$ contains for each step s, the state of the different interaction protocols and agents. For instance, in amWater one possible initial state (S_0) of an execution of the *Auction protocol* in a particular auction scene could be: the auction scene is opened but the auction has not yet started; there are two agents within the scene: one enacting the role of *market facilitator* and one acting as *basin authority*. Similarly, $Norms_s$ stands for the execution state of norms at step s. This includes: i) the set of active norms (when one auction is running, only bidders in the last iteration are allowed to bid), ii) the set of pending obligations as a result of the applied norms (when the auction is paused, the $winner$(s) has to request a water quantity), iii) the list of norm violations (on the auction's pause time-out, the $winner$(s) has not requested any quantity), or iv) the list of applied sanctions due to norm violations (minimum quantity is assigned to the $winner$(s) because s/he has not requested any quantity).

– $Goals_s$ represents the degree of fulfilment of each organisational goal and the degree of satisfaction of agents' goals. While the former can be computed by the organisation, the latter can only be gathered by asking the agents.

– AgP_s are the values of the properties of agents, where the property j of agent i ($agP_{i,j}$) is the attribute-value pair: $\langle att_{i,j}, value_{i,j} \rangle$. For example in amWater, the property $quantity$ of the k-th water right of a seller ($seller.wrs(k).quantity$) and of the l-th buyer's water right ($buyer.wrs(l)$) change when a *basin authority* validates an agreement on transfer quantity qt of water from $seller.wrs(k)$ to $buyer.wrs(l)$. These two operations are performed atomically: $seller.wrs(k).quantity = seller.wrs(k).quantity - qt$; $buyer.wrs(l).quantity = buyer.wrs(l).quantity + qt$.

– $EnvP_s$ are relevant values of organisational and/or environmental properties. The list of opened negotiations and the items they are going to negotiate constitute some examples in amWater.

The transitions between execution states of the organisation are mostly driven by actions. We define agents' possible actions (A) at step s (A_s) as :

$$A_s = \{a_1, \ldots, a_n\} \quad (2)$$

where n is the number of agents of the institution and a_i is the action performed by the agent with unique identifier i. We assign the $skip$ action to a_i ($a_i = skip$) when agent i did not perform any action.

S and A denote the sequence of different consecutive execution states of the organisation.In this way, we finally define the *Organisational Trace* ($Trac$) as:

$$Trac = \{(S_0, A_0), \ldots, (S_k, A_k), \ldots, (S_c)\} \quad (3)$$

where S_0 is the initial state, S_c is the current state and in general A_k is the set of actions that take place at state S_k and step k. Scalability and distribution should be taken into account when managing the presented $Trac$, but this is out of scope in this paper which only pretends to formalise it.

As an example in amWater, let imagine that one particular auction scene is opened at current S_c, but the auction has not yet started (*auction.state* = *stopped*). It just starts when

the market facilitator says a message with the content *"start round"*. Hence, current state changes (S_c = *{auction.state = running}*) as a consequence of the market facilitator's action A_c = *{send_message ("start round")}*.

4. Assistance Infrastructure

With the aim of assisting –further than enabling– agent coordination, we propose a new set of assistance services that could be incorporated in MAS infrastructures. The services may concern: personal information, only known by the agent; group information, that corresponds to information about agents' groups; and/or global information, information about the whole system.

4.1. Assistance Services

There are three main issues an agent should face to effectively participate in a given organisation: first, the agent only has access to partial information about the actual organisation execution state and its environment; second, the information gathered by the agent should be processed in order to obtain useful knowledge; third, agents have to reason about the coordination model defined by the organisation. Therefore, agents should elaborate plans to achieve their individual goals. On one hand, when programming agents, a developer is required to implement such a planning tasks. On the other hand, fulfilling these tasks by human agents becomes even more complex because of limited speed of human information processing system.

We state that *Assistance services* improve both human and software agents participation in the organisation because: first, they have access to the *Organisational Trace*, $Trac$; second, services are designed to offer complex processes; third, they reason about the coordination model in line with agents' goals.

In addition to services offering basic organisational information, we propose the following services: (1) providing agents with refined *information* to participate in the MAS, (2) *justifying* action consequences or effective constraints, (3) giving *advice* to agents and/or (4) *estimating* action consequences. These services may be provided in a distributed way under request and/or by subscription. Different frequencies of subscriptions could be: each time the information changes, e.g. when a new water right has been registered to transfer; regular frequency, e.g. at the beginning of each period of the irrigation campaign; only when it is interesting for the agent, e.g. the first time entering the organisation the agent receives a "welcome pack"; and never (e.g. subscription disabled).

4.1.1. Information

This service provides, besides basic organisational information offered by most infrastructures, complex processed information. We can assume that the infrastructure informs agents proactively (equation 4) or upon a request (equation 5). In both cases, we assume in this definition that the infrastructure sends information about the organisation specification (Org) and its execution state ($Trac$):

$$Information : Org \times Trac \rightarrow I \tag{4}$$

$$Information : Req \times Org \times Trac \rightarrow I \tag{5}$$

where Req is any possible request for information an agent can send (e. g. current norms, statistics about specific protocols or environmental property values). The provided or accessible information for an agent depends on its role, properties and its context. For instance, different roles may have access to different information or an agent may only receive information about those interaction protocols he is taking part on.

In amWater scenario, to the irrigator's question *"How many water rights did other irrigators buy?"*, some possible answers could be: *"The market is very active, each irrigator did buy 100 m^3 in average (Average)"*; or *"Here you have the list of (public) validated market transactions (Detail)"*

4.1.2. Justification

This service justifies *the consequences* of actions that agents perform. These consequences depend on the action itself, the current social conventions and the current context. For instance, depending on the enforcement policy, an action can be filtered out or performed with extra consequences –e.g. new obligations–. In both cases we suggest to provide a justification to the agent that performed it. Equation 6 defines this service.

$$Justification : A \times Org \times Trac \rightarrow J \tag{6}$$

where J stands for the set of possible justifications. Hence, we formalise this service as a function that given an agent action, the organisation and its trace, it returns the corresponding justification. The justification provided by the infrastructure may consist on: 1) the involved norm(s) (e. g., if an auction is running then only buyers can bid); 2) the involved norm(s) and current state values (e. g. 1) and auction is not running);

4.1.3. Advice

This service provides agents with a set of alternative plans (P) —i.e. action sequences: $P = \{a_1, \ldots, a_m\}$, where m is the number of actions conforming the plan. The way to create an *advice* can be as simple as indicating what actions other agents have performed on the same situation. Or it can be as complex as planning possible actions given current state, regulations –e.g. social conventions– and goals. We define two specialisations of advice: Imitation and Planning. [2]

Imitation. We can provide some plans defined as the most common action, A, carried out by other agents facing the same situation. Note that this service does not require that the agent reveal its goals. An advice in this case could be defined as:

$$Advice : Org \times Trac \rightarrow A \tag{7}$$

In amWater, one buyer may ask *"How much should I pay for the water rights I am interested in?"* and the infrastructure could answer with the advice: *other agents paid (in average) 10 monetary units per m^3*.

Planning. In order to offer a planning advice, we assume that the agent provides a goal ($IndGoal$) it wants to achieve. This service is defined as:

$$Advice : IndGoal \times Org \times Trac \rightarrow P^N \tag{8}$$

where P^N is a set of alternative plans aimed to achieve goal $IndGoal$, and N is the number of plans conforming the advice. The advice provides plans with the utility of the agent maximised.

For instance, in amWater, the first time an agent enters the institution as a buyer, in the Waiting scene s/he may request for an advice on *"How can I purchase rights?"*. One possible advice may be the following sequence of actions: 1) leave the waiting scene; 2) go to the auction scene number 14; 3) wait until auction starts; 4) bid.

[2] It is important to notice, though, that plans can fail due to the actions performed by other agents or to changes happened in the environment. Therefore, more sophisticated plans may be produced by having into account other possible agent's actions, advices given to them and/or environmental changes. But these three options could make really complex the problem to solve. In first attempt, we expect to just use current state and norms as input parameters of the service.

4.1.4. Estimation

An estimation service predicts the organisational state reached if an agent's indicated action is executed.

$$Estimation : IndA \times Org \times S_c \rightarrow E \tag{9}$$

where E stands for the set of possible estimations. The agent provides an action s/he wants to execute, and the infrastructure uses the organisation definition and the current state (S_c) to estimate its consequences. We provide three different possibilities for the E: 1) $Boolean$. True meaning that the action is valid, while false that it is invalid; 2) $Boolean \times J \times Consequences$. The boolean has the previous meaning, J stands for a justification and *Consequences* stands for the consequences of executing the action. Regarding *Consequences*, one possibility is to assume that it could be a list of obligations the agent will acquire if the action is executed (when the action is valid) or the list of sanctions (when it is invalid); 3) An estimation of the system state after executing the action (e. g., expected values of properties or expected degree of goal achievement).

Estimation may work as a decision support for agents, which can directly focus on evaluating action consequences instead of directly evaluating social conventions. In fact, an agent could use a learning mechanism to acquire empirical knowledge about social conventions instead of analysing their specifications.

Some examples of the three different E in amWater could be: Q1:*"Can I enter to the auction scene number 14?"* E1: *Yes / No.* Q2:*"Can I request 40 m^3 of water?"* E2: *Yes, because you won, then you have to pay 400 m. u. / No, because auction is still running.* Q3:*"What If I bid at current price?"* E3: *You may win / The price will be increased.*

4.2. Assisted Groups

Groups are usually defined in MAS as sets of agent aggregation, where each agent is part of at least one group and it may be member of several groups at the same time so that groups can freely overlap [11]. This definition of MAS' groups is based on organisational activities. Our proposition is in the same line, but based on agent properties related to their services of interest.

Because we are considering open MAS, agents are heterogeneous. Their goals, preferences and interests may vary and are unknown for the system. Agents will be specially interested in assistance services regarding her/his goals. Therefore, agents can be grouped depending on their interests. As a consequence, agents in the same group could not participate in the very same activity. We define an *Assisted Group* ($AssGrp$) as:

$$AssGrp = \mathcal{F}\left(Ag, AgP\right) \tag{10}$$

$\mathcal{F}\left(Ag, AgP\right)$ is a boolean function, that is, it will return true if the agent belongs to the group and false otherwise. Ag is the set of agents and AgP is the set of their properties. The properties and their necessary conditions to be member of a group should be identified at design time with a basic expression language. Each different definition of group will offer a different partitioning of the agents interacting in the domain.

As previously mentioned in amWater interconnected rights are located within the same area. A general goal of any irrigator entering the institution will be to transfer water between their owned rights and other interconnected rights, by buying or selling them. Interconnected rights are identified by the area they are located in. Therefore, most irrigators will be specially interested in services concerning the area of their rights. As a consequence, they will generally request for such services to the *Assistance Layer*. Then, we group irrigators by the property $areas$ that is the set of areas where they own lands.

Therefore, function $AssGrp$ in this particular domain is defined as: $AssGrp\,(area) = \{x \in Ag, x : irrigator | area \in x.areas\}$ [3]

4.3. Assistants

Personal Assistants. One Personal Assistant (PA) may provide direct and sole support to one agent of the open MAS. PA could observe the organisational trace ($Trac$) so as to deploy the *Assistance Services*.

Advice and *Estimation* services (see section 4.1) could be provided in line with agents' goals, i.e., maximising the agent's utility. As open MAS, agent's goals are information only known by the agent. Therefore, an agent should reveal her/his personal information to its PA in order to be adequately assisted. We stress that it is the assisted agent's decision which personal information s/he communicates to the PA. Obviously, the more relevant information revealed, the better *Advice* and *Estimation* services will be provided. We propose to establish a private communication channel between the PA and its assisted agent in order to preserve the privacy of the information in the communications. To ensure the use of private information in the defined terms and conditions, a service contract may be signed between assistants and agents. Therefore, we conclude that agents should feel confidence using PAs.

Group Assistants. We propose one Group Assistant (GA) to provide services specialised in one *Assisted Group* ($AssGrp$). The GA could implements complex processes using information concerning its $AssGrp$. This information could be provided by accessing to the organisational trace ($Trac$). Moreover, as discussed above, some services need agent's personal information in order to be performed. We propose that the agent has the decision to reveal its personal information to the GA, and the GA should not reveal nor store such information.

We extract the next advantages from introducing a GA: 1) it provides services interesting to its $AssGrp$ by interacting with all related PAs; 2) GA keeps $Trac$ information relevant to the group. 3) PA computing time and communication overload may be alleviated. 4) The services provided by the GA preserve the privacy policy.

5. Conclusions and future work

In this paper we propose an assistance infrastructure to support agents participation in an Open MAS. This infrastructure is composed by two types of assistants agents and four assistance services. These services offer, in addition to basic organisational information, more complex services such as 1) elaborated *Information*, e. g. specific statistics; 2) *Justification* of the consequences of a performed action; 3) giving *Advice*, e. g. providing a plan to achieve agent's goals; and 4) *Estimation* of the consequences of performing an agent's action prior to its execution. A *Personal Assistant* may provide direct and sole support to one agent of the open MAS. A *Group Assistant* performs complex data processes concerning information about one *Assisted group* which groups agents based on their properties regarding their services of interest. As intended work, we will complete

[3] Note that in this definition of groups, one particular *irrigator* belongs to at least one area and s/he may belong to more than one area because s/he could have lands in different areas. Therefore, one agent is part of one or more groups and groups may overlap, as defined in Ferber et al. [11]. Moreover, there may be one *irrigator* in the institution with lands in several areas (so potentially belonging to different groups) that only wants to be assisted on one particular area, so normally s/he only will request support to such area services. As a difference with [11], agents within the same group can be located in a different scene when they are using the same service.

the formalisation of the infrastructure by further specifying the elements proposed in this paper and providing an architecture that enacts them in an open OCMAS.

In order to illustrate our approach we use an open OCMAS example scenario that implements an electronic market of water rights. Nevertheless the proposed services still require being included in this example. We leave this as future work.

References

[1] J. Arcos, M. Esteva, P. Noriega, J. Rodríguez-Aguilar, and C. Sierra. An integrated development environment for electronic institutions. In *Software Agent-Based Applications, Platforms and Development Kits*, pages 121–142. 2005.

[2] A. Bogdanovych, M. Esteva, S. Simoff, C. Sierra, and H. Berger. A methodology for developing multiagent systems as 3d electronic institutions. volume 4951 of *Lecture Notes in Computer Science*, pages 103–117. Springer Berlin / Heidelberg, 2008.

[3] E. Bou, M. López-Sánchez, and J. Rodríguez-Aguilar. Towards self-configuration in autonomic electronic institutions. volume 4386 of *Lecture Notes in Computer Science*, pages 229–244. 2007.

[4] E. Bou, M. López-Sánchez, and J. Rodríguez-Aguilar. *Autonomic Electronic Institutions' Self-Adaptation in Heterogeneous Agent Societies*, volume 5368, pages 18–35. Springer, 2009.

[5] J. Campos, M. Esteva, M. López-Sánchez, J. Morales, and M. Salamó. Organisational adaptation of multi-agent systems in a peer-to-peer scenario. *Computing*, 91:169–215, 2011.

[6] J. Campos, M. López-Sánchez, and M. Esteva. Coordination support in multi-agent systems. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, pages 1301–1302, 2009.

[7] R. Centeno and H. Billhardt. Adaptive regulation of open mas: an incentive mechanism based on online modifications of the environment (extended abstract). In *Proceedings of The 10th International Conference on Autonomous Agents and Multiagent Systems*, pages 1243–1244, 2011.

[8] R. Centeno, H. Billhardt, R. Hermoso, and S. Ossowski. Organising mas: a formal model based on organisational mechanisms. In *Proceedings of the 2009 ACM symposium on Applied Computing*, pages 740–746, 2009.

[9] H. Chalupsky, Y. Gil, C. A. Knoblock, K. Lerman, J. Oh, D. V. Pynadath, T. A. Russ, and M. Tambe. Electric elves: Applying agent technology to support human organizations. In *Proceedings of the Thirteenth Conference on Innovative Applications of Artificial Intelligence Conference*, pages 51–58. AAAI Press, 2001.

[10] M. Esteva. *Electronic institutions. from specification to development*. PhD thesis, Universitat Politecnica de Catalunya, 2003.

[11] J. Ferber and O. Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. *Multi-Agent Systems, International Conference on*, pages 128–135, 1998.

[12] J. Ferber, O. Gutknecht, and F. Michel. From agents to organizations: An organizational view of multiagent systems. In *Agent-Oriented Software Engineering IV*, volume 2935, pages 443–459. 2004.

[13] A. Garrido, A. Giret, and P. Noriega. mwater: a sandbox for agreement technologies. In *Proceeding of the 2009 conference on Artificial Intelligence Research and Development: Proceedings of the 12th International Conference of the Catalan Association for Artificial Intelligence*, pages 252–261, 2009.

[14] J. Hübner, J. Sichman, and O. Boissier. $\mathcal{S}-\mathcal{M}oise^{+}$: A middleware for developing organised multiagent systems. In *Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems*, volume 3913, pages 64–77. 2006.

[15] N. R. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1:7–38, 1998.

[16] I. J. Jureta, M. J. Kollingbaum, S. Faulkner, J. Mylopoulos, and K. Sycara. Requirements-driven contracting for open and norm-regulated multi-agent systems. Technical report, 2007.

[17] J. Oh, F. Meneguzzi, K. Sycara, and T. J. Norman. Prognostic agent assistance for norm-compliant coalition planning. In *The Second International Workshop on INFRASTRUCTURES AND TOOLS FOR MULTIAGENT SYSTEMS*, pages 126–140, 2011.

[18] S. Okamoto, K. Sycara, and P. Scerri. Personal assistants for human organizations. In V. Dignum, editor, *Organizations in Multi-Agent Systems*. 2009.

[19] T. Trescak, M. Esteva, and I. Rodriguez. Vixee an innovative communication infrastructure for virtual institutions (extended abstract). In *Proceedings of The 10th International Conference on Autonomous Agents and Multiagent Systems*, pages 1131–1132, 2011.

On the Modularity of Industrial SAT Instances[1]

Carlos Ansótegui [a,2] Jordi Levy [b]
[a] *Universitat de Lleida (DIEI, UdL)*
[b] *Artificial Intelligence Research Institute (IIIA, CSIC)*

Abstract. Learning, re-starting and other techniques of modern SAT solvers have been shown efficient when solving SAT instances from industrial application. The ability to exploit the structure of these instances has been proposed as the responsible of such success. Here we study the modularity of some of these instances, used in the latest SAT competitions. Using a simple label propagation algorithm we show that the community structure of most of these SAT instances can be identified very efficiently. We also discuss how this structure may be used to speed up SAT solvers.

Keywords. Satisfiability, graph theory, modularity

Introduction

In recent years, SAT solvers efficiency solving industrial instances has undergone a great advance, mainly motivated by the introduction of lazy data-structures, learning mechanisms and activity-based heuristics [7,14]. This improvement is not shown when dealing with randomly generated SAT instances. The reason for this difference seems to be the existence of a structure in industrial instances [21].

In parallel, there have been significant advances in our understanding of complex networks, a subject that has focused the attention of statistical physicists. The introduction of these network analysis techniques will help us understand the nature of SAT instances, and will contribute to further improve the efficiency of SAT solvers. Watts and Strogatz [20] introduce the notion of *small word*, the first model of complex networks, as an alternative to the classical random graph models. Walsh [19] analyzes the small word topology of many graphs associated with search problems in AI. He also shows that the cost of solving these search problems can have a *heavy-tailed distribution*. Gomes et al. [10,11] propose the use of *randomization* and *rapid restart* techniques to prevent solvers to fall on the long tail of such kinds of distributions.

The notion of structure has been addressed in previous work [10,12,9,13]. The closest work to our contribution is [18], but it uses the notion of modularity in a different

[1]Research partially founded by the CICyT research projects TIN2010-20967-C04-01/03 and TIN2009-14704-C03-01 .

[2]Corresponding Author: Departament d'Informàtica i Enginyeria Industrial, Jaume II 69, 25001 Lleida, Spain; E-mail: carlos@diei.udl.cat

sense. Also, in [1] some techniques are proposed to reason with multiple knowledge bases that overlap in content. In particular, they discuss strategies to induce a partitioning of the axioms, that will help to improve the efficiency of reasoning. In [4], it is shown that many SAT instances can be decomposed into connected components, and how to handle them within a SAT solver. They also discuss how what they call *component structure* can be used to improve the performance of SAT solvers.

In this paper we propose the use of techniques for detecting the *community structure* of SAT instances. In particular, we apply the notion of *modularity* [15] to detect these communities. We also discuss how existing conflict directed clause learning algorithms and activity-based heuristics already take advantage, *indirectly*, of this community structure. Activity-based heuristics [14] rely on the idea of giving higher priority to the variables that are involved in (recent) conflicts. By focusing on a sub-space, the covered spaces tend to coalesce, and there are more opportunities for resolution since most of the variables are common.

1. Preliminaries

Given a set of Boolean variables $X = \{x_1, \ldots, x_n\}$, a *literal* is an expression of the form x_i or $\neg x_i$. A *clause* c of length s is a disjunction of s literals, $l_1 \vee \ldots \vee l_s$. We say that s is the size of c, noted $|c|$, and that $x \in c$, if c contains the literal x or $\neg x$. A *CNF formula* or *SAT instance* of length t is a conjunction of t clauses, $c_1 \wedge \ldots \wedge c_t$.

An (undirected) graph is a pair (V, w) where V is a set of vertices and $w : V \times V \to \mathbb{R}^+$ satisfies $w(x, y) = w(y, x)$. This definition generalizes the classical notion of graph (V, E), where $E \subseteq V \times V$, by taking $w(x, y) = 1$ if $(x, y) \in E$ and $w(x, y) = 0$ otherwise. The degree of a vertex x is defined as $\deg(x) = \sum_{y \in V} w(x, y)$. A bipartite graph is a tuple (V_1, V_2, w) where $w : V_1 \times V_2 \to \mathbb{R}^+$.

Given a SAT instance, we construct two graphs, following two models. In the Variable Incidence Graph model (VIG, for short), vertices represent variables, and edges represent the existence of a clause relating two variables. A clause $x_1 \vee \ldots \vee x_n$ results into $\binom{n}{2}$ edges, one for every pair of variables. Notice also that there can be more than one clause relating two given variables. To preserve this information we put a higher weight on edges connecting variables related by more clauses. Moreover, to give the same relevance to all clauses, we ponderate the contribution of a clause to an edge by $1/\binom{n}{2}$. In this way, the sum of the weights of the edges generated by a clause is always one. In the Clause-Variable Incidence Graph model (CVIG, for short), vertices represent either variables or clauses, and edges represent the occurrence of a variable in a clause. SAT instances use to be simplified: there are no two occurrences of a literal (because $x \vee x \vee c$ is equivalent to $x \vee c$) nor two occurrences of the same variable and opposite sign (because $x \vee \neg x \vee c$ is a tautology). Therefore, in the CVIG model, edges have weight one.

Definition 1 (Variable Incidence Graph (VIG)) *Given a SAT instance Γ over the set of variables X, its variable incidence graph is a graph (X, w) with set of vertices the set of Boolean variables, and weight function:*

$$w(x, y) = \sum_{\substack{c \in \Gamma \\ x, y \in c}} \frac{1}{\binom{|c|}{2}}$$

Definition 2 (Clause-Variable Incidence Graph (CVIG)) *Given a SAT instance Γ over the set of variables X, its clause-variable incidence graph is a bipartite graph $(X, \{c \mid c \in \Gamma\}, w)$, with vertices the set of variables and the set of clauses, and weight function:*

$$w(x,c) = \begin{cases} 1 \text{ if } x \in c \\ 0 \text{ otherwise} \end{cases}$$

2. Modularity in Large-Scale Graphs

To analyze the structure of a SAT instance we will use the notion of *modularity* introduced by [16]. This property is defined for a graph and a specific *partition* of its vertices into *communities*, and measures the adequacy of the partition in the sense that most of the edges are within a community and few of them connect vertices of distinct communities. The modularity of a graph is then the maximal modularity for all possible partitions of its vertices. Obviously, measured this way, the optimal would be obtained putting all vertices in the same community. To avoid this problem, Newman and Girvan define modularity as the fraction of edges connecting vertices of the same community minus the expected fraction of edges for a random graph with the same number of vertices and same degree.

Definition 3 (Modularity of a Graph) *Given a graph $G = (V, w)$ and a partition $P = \{P_1, \ldots, P_n\}$ of its vertices, we define their* modularity *as*

$$Q = \sum_{P_i \in P} \frac{\sum\limits_{x,y \in P_i} w(x,y)}{\sum\limits_{x,y \in V} w(x,y)} - \left(\frac{\sum\limits_{x \in P_i} deg(x)}{\sum\limits_{x \in V} deg(x)} \right)^2$$

We call the first term of this formula the inner edges fraction*, IEF for short, and the second term the* expected inner edges fraction*, IEF^e for short. Then, $Q = IEF - IEF^e$.*

The (optimal) modularity *of a graph is the maximal modularity, for any possible partition of its vertices.*

Since the IEF and the IEF^e of a graph are both in the range $[0, 1]$, and, for the partition given by a single community, both have value 1, the optimal modularity of graph will be in the range $[0, 1]$. A value of 0.3 is considered as a high evidence of community structure in a graph.

There has not been an agreement on the definition of modularity for bipartite graphs. Here we will use the notion proposed by [2] that extends Newman and Girvan's definition by restricting the random graphs used in the computation of the IEF^{e}to be bipartite. In this definition, communities may contain vertices of V_1 and of V_2.

Definition 4 (Modularity of a Bipartite Graph) *Given a graph $G = (V_1, V_2, w)$ and a partition $P = \{P_1, \ldots, P_n\}$ of its vertices, we define their* modularity *as*

$$Q = \sum_{P_i \in P} \frac{\sum\limits_{\substack{x \in P_i \cap V_1 \\ y \in P_i \cap V_2}} w(x,y)}{\sum\limits_{\substack{x \in V_1 \\ y \in V_2}} w(x,y)} - \frac{\sum\limits_{x \in P_i \cap V_1} deg(x)}{\sum\limits_{x \in V_1} deg(x)} \cdot \frac{\sum\limits_{y \in P_i \cap V_2} deg(y)}{\sum\limits_{y \in V_2} deg(y)}$$

There exist a wide variety of algorithms for computing the modularity of a graph. Moreover, there exist alternative notions and definitions of modularity for analyzing the community structure of a network. See [8] for a survey in the field. The decision version of modularity maximization is NP-complete [5]. All the modularity-based algorithms proposed in the literature return an approximated lower bound for the modularity. They include greedy methods, methods based on simulated annealing, on spectral analysis of graphs, etc. Most of them have a complexity that make them inadequate to study the structure of an industrial SAT instance. There are two algorithms specially designed to deal with large-scale networks: the greedy algorithms for modularity optimization [15,6], and a label propagation-based algorithm [17].

The first algorithm for modularity maximization is a *greedy method* of Newman [15]. This algorithm starts by assigning every vertex to a distinct community. Then, it proceeds by joining the pair of communities that result in a bigger increase of the modularity value. The algorithm finishes when no community joining results in an increase of the modularity. In other words, it is a greedy gradient-guided optimization algorithm. The algorithm may also return a dendogram of the successive partitions found. Obviously, the obtained partition may be a local maximum. In [6] the data structures used in this basic algorithm are optimized, using among other data structures for sparse matrices. The complexity of this refined algorithm is $\mathcal{O}(m\, d\, \log n)$, where d is the depth of the dendogram (i.e. the number of joining steps), m the number of edges and n the number of vertices. They argue that d may be approximated by $\log n$, assuming that the dendogram is a balanced tree, and the sizes of the communities are similar. However, this is not true for the graphs we have analyzed, where the sizes of the communities are not homogeneous. This algorithm has not been able to finish, for none of our SAT instances, with a run-time limit of one hour.

An alternative algorithm is the *Label Propagation Algorithm* proposed by [17] (see Figure 1). Initially, all vertices are assigned to a distinct label, e.g., its identifier. Then, the algorithm proceeds by re-assigning to every vertex the label that is more frequent among its neighbors. The procedure ends when every vertex is assigned a label that is maximal among its neighbors. The order in which the vertices update their labels in every iteration is chosen randomly. In case of a tie between maximal labels, the winning label is also chosen randomly. The algorithm returns the partition defined by the vertices sharing the same label. The label propagation algorithm has a near linear complexity. However, it has been shown experimentally that the partitions it computes have a worse modularity than the partitions computed by the Newman's greedy algorithm.

3. Modularity of SAT Instances

We have computed the modularity of the SAT instances used in the 2010 SAT Race Finals (see `http://baldur.iti.uka.de/sat-race-2010/`). They are 100 in-

Input: $G = (X, w)$
Output: a labelling for X
for $x \in X$ **do** label[x] := x; freq[x] := 0 **endfor**
do
  ord := shuffle(X)
  changes := false
  for i:=0 **to** $|X|$ **do**
    (l, f) := most_freq_label(ord[i],neighbors(ord[i]))
    changes := changes $\vee$ $f >$ freq[ord[i]]
    label[ord[i]] := l
    freq[ord[i]] := f
  endfor
while changes
return label

function most_freq_label(v, N)
  L :={label[v] | $v \in N$}
  for $l \in L$ **do**
    freq[l] := $\sum_{\substack{v' \in N \\ l = \text{label}[v']}} w(v, v')$
  Max := $\{l \in L \mid \text{freq}[l] = \max\{\text{freq}[l] \mid l \in L\}\}$
return random_choose(Max)

Figure 1. Label Propagation Algorithm. The function most_freq_label returns the label that is most frequent among a set of vertices. In case of tie, it randomly chooses one of the maximal labels.

stances grouped into 16 families. These families are also classified as cryptography, hardware verification, software verification and mixed, according to their application area. All instances are *industrial*, in the sense that their solubility has an industrial or practical application. However, they are expected to show a distinct nature.

Our experiments have been run on a cluster with the following specifications. Operating System: Rocks Cluster 4.0.0 Linux 2.6.9, Processor: AMD Opteron 248 Processor, 2 GHz, Memory: 1 GB, and Compiler GCC 3.4.3.

We have observed that all instances of the same family have a similar modularity. Therefore, in Table 1, we only show average values, being their standard deviation quite small (smaller that 10% of the mean in almost all cases).

As one could expect, the time needed to compute the modularity in the CVIG model is bigger than the time for the VIG model, since CVIG graphs are bigger. However, the number of iterations of the algorithm is also bigger in the CVIG model.

If we take $Q > 0.3$ as an evidence of community structure, we can conclude that most families are modular in the VIG model, and that all but one are modular in the CVIG model. Except for the bioinf family, the modularity is always bigger in the CVIG model than in the VIG model. In some families, like mizh, fuhs or nec, the modularity for VIG is meaningless, whereas the CVIG graph shows a clear community structure. It could be concluded that the loss of information, during the projection of the bipartite CVIG graph into the set of variables, may destroy part of the modular structure. However, this is not completely true. Suppose that the instance has no modular structure at all, but all clauses

Family (#instanc.)		Variable Incid. Graph (VIG)							Clause-Variable Incid. Graph (CVIG)						
		time	IEF	IEF^e	Q	$\|P\|$	larg.	iter.	time	IEF	IEF^e	Q	$\|P\|$	larg.	iter.
cripto.	desgen(4)	7	0.89	0.01	**0.88**	517	0.01	34	15	0.77	0.00	**0.77**	2752	0.01	28
	md5gen(3)	6	0.61	0.00	**0.61**	7151	0.00	15	43	0.78	0.00	**0.78**	6934	0.00	32
	mizh(8)	2	1.00	1.00	0.00	21	1.00	4	52	0.78	0.10	**0.69**	4505	0.30	33
hard. ver.	ibm(4)	13	0.80	0.00	**0.80**	2681	0.01	10	51	0.79	0.00	**0.79**	8839	0.00	18
	manolios(16)	10	0.97	0.71	0.26	44	0.82	9	100	0.76	0.00	**0.76**	5115	0.01	35
	velev (10)	(3)	0.90	0.52	**0.38**	70	0.65	10							
mixed	anbulagan(8)	31	0.56	0.00	**0.56**	36745	0.00	11	88	0.87	0.00	**0.87**	13875	0.00	18
	bioinf(6)	8	0.80	0.18	**0.62**	172	0.38	4	37	0.82	0.37	**0.46**	2147	0.43	30
	diagnosis(4)	62	0.63	0.00	**0.63**	16372	0.02	14	198	0.74	0.00	**0.74**	31999	0.00	25
	grieu(3)	0	1.00	1.00	0.00	1.0	1.00	2	9	0.97	0.92	0.05	1.7	0.96	13
	jarvisalo(1)	0	0.59	0.01	**0.57**	260	0.05	8	0	0.73	0.00	**0.72**	294	0.01	12
	palacios(3)	134	0.96	0.62	**0.34**	2117	0.66	66	269	0.81	0.10	**0.72**	2853	0.17	49
soft. ver.	babic(2)	61	0.70	0.02	**0.68**	34033	0.08	54	379	0.72	0.01	**0.71**	61577	0.05	86
	bitverif(5)	78	0.97	0.58	**0.39**	83	0.64	96	363	0.82	0.02	**0.80**	9145	0.05	199
	fuhs(4)	8	0.93	0.76	0.17	379	0.79	25	5	0.71	0.06	**0.64**	5747	0.13	13
	nec(10)	207	0.99	0.87	0.12	372	0.93	22	882	0.80	0.02	**0.78**	31914	0.02	114

Table 1. Computation of the modularity of the 2010 SAT Race instances, using the Label Propagation Algorithm. Time (in seconds) is the CPU time needed to propagate labels (excluding parsing the formula and constructing the graph). $Q = \text{IEF} - \text{IEF}^e$ is the modularity. $|P|$ is the number of communities, and $larg$ the fraction of vertices in the largest community. Iter is the number of iterations of the algorithm.

are binary. We can construct a partition as follows: put every variable into a distinct community, and every clause into the same community of one of its variables. Using this partition, half of the edges will be internal, i.e. IEF $= 0.5$, IEF^e will be nearly zero, and $Q \approx 0.5$. Therefore, we have to take into account that using Barber's modularity definition for bipartite graphs, as we do, if vertex degrees are small, modularity can be quite big compared with Newman's modularity.

Finally, we also report results on the number of communities ($|P|$) and the fraction of vertices belonging to the largest community ($larg$). If all communities have a similar size, then $larg \approx 1/|P|$. In some cases, like palacios and mizh, we have $|P| \gg 1/larg$. This means that the community structure corresponds to a big (or some) big central communities surrounded by a multitude of small communities. The existence of a big community implies an expected inner fraction close to one, hence a modularity close to zero.

We have also conducted a study of the modularity of 100 random 3-CNF SAT instances. For the VIG model, we have observed that, with 10^5 variables and $4.25 \cdot 10^5$ clauses (i.e. around the phase transition point) the average modularity is $Q = 0$, the average number of communities $|P| = 1.22$ and largest size fraction $larg = 1$. Therefore, almost surely the algorithm collapses into a single community. For under-constrained problems with 10^5 variables and 10^5 clauses, the average modularity is $Q = 0.025$. In this case there is a largest community with 92% of the vertices, on average, although there are also 5001 additional communities. For the CVIG model, in the phase transition

Family	Variable Incid. Graph							Clause-Variable Incid. Graph						
	orig.	first 100 learned			all learned			orig	first 100 learned			all learned		
	Q	IEF	IEF^e	Q	IEF	IEF^e	Q	Q	IEF	IEF^e	Q	IEF	IEF^e	Q
desgen(1)	**0.89**	0.77	0.03	**0.74**	0.12	0.04	0.08	**0.77**	0.33	0.05	0.28	0.14	0.04	0.09
md5gen(1)	**0.61**	0.76	0.02	**0.74**	0.03	0.01	0.02	**0.78**	0.99	0.03	**0.96**	0.03	0.01	0.02
ibm(2)	**0.84**	0.70	0.11	**0.60**	0.48	0.01	**0.47**	**0.81**	0.70	0.11	**0.58**	0.29	0.00	0.29
manolios(10)	0.21	0.87	0.84	0.04	0.80	0.71	0.10	**0.76**	0.13	0.02	0.11	0.10	0.01	0.09
anbulagan(2)	**0.56**	0.18	0.02	0.16	0.02	0.01	0.01	**0.87**	0.10	0.01	0.10	0.06	0.02	0.04
bioinf(4)	**0.62**	0.57	0.12	**0.46**	0.42	0.36	0.06	**0.68**	0.77	0.08	**0.69**	0.24	0.09	0.15
grieu(1)	0.00	1.00	1.00	0.00	1.00	1.00	0.00	0.00	1.00	1.00	0.00	1.00	1.00	0.00
babic(2)	**0.68**	0.84	0.48	**0.36**	0.84	0.48	**0.36**	**0.71**	0.55	0.22	**0.33**	0.55	0.22	**0.33**
fuhs(1)	**0.66**	0.67	0.08	**0.59**	0.24	0.10	0.14	**0.71**	0.80	0.02	**0.78**	0.09	0.01	0.07
nec(10)	0.12	0.89	0.88	0.01	0.96	0.84	0.12	**0.78**	0.73	0.24	**0.49**	0.70	0.46	0.24

Table 2. Modularity of the graph generated by the formula containing the first 100 learned clauses, and all learned clauses. In the first column we show the modularity of the original formula.

Family	Variable Incid. Graph							Clause-Variable Incid. Graph						
	orig	first 100 learned			all learned			orig.	first 100 learned			all learned		
	Q	IEF	IEF^e	Q	IEF	IEF^e	Q	Q	IEF	IEF^e	Q	IEF	IEF^e	Q
md5gen(1)	**0.61**	0.76	0.02	**0.74**	0.03	0.01	0.02	**0.78**	0.99	0.03	**0.96**	0.03	0.01	0.02
ibm(2)	**0.84**	0.56	0.06	**0.50**	0.47	0.04	**0.43**	**0.81**	0.66	0.07	**0.58**	0.46	0.05	**0.42**
anbulagan(1)	**0.55**	0.04	0.00	0.03	0.01	0.01	0.00	**0.87**	0.14	0.01	0.13	0.07	0.02	0.05
manolios(9)	0.20	0.86	0.79	0.07	0.85	0.75	0.10	**0.76**	0.29	0.03	0.27	0.16	0.01	0.15
nec(10)	0.12	0.90	0.85	0.05	0.90	0.85	0.05	**0.78**	0.65	0.35	**0.30**	0.50	0.28	0.22

Table 3. Modularity of the learned clauses that have been contributed to prove the unsatisfiability of the original formula. Like in Table 2 we show results for the first 100 learned clauses, and for all clauses.

point, we get $Q = 0.007$, $|P| = 1.29$ and $larg = 0.99$. For 10^5 variables and clauses, we get $Q = 0.12$, $larg = 0.86$ and $|P| = 6547$.

Notice that in the CVIG model for random 3-CNF formulas, using the same argument as above, i.e. assigning a distinct community to each variable, and to each clause the community of one of its variables, we get a graph partition with $Q \approx 0.33$. Therefore, in these examples the label propagation algorithm does not always compute the maximal modularity, and tends to collapse all communities into a single one. However, analyzing the results for each one of the formulas independently, we observe that in some of them the algorithm is able to compute this maximal modularity. Therefore, it would make sense to run the algorithm, with distinct seeds, and take the partition with maximal modularity.

[3] Velev instances are huge. We have to compute their modularity in a computer with more memory. In the case of the CVIG model, even this second computer is unable to compute the partition.

4. Modularity of the Learned Clauses

Most modern SAT solvers, based on variants of the DPLL schema, transform the formula during the proof or the satisfying assignment search. Therefore, the natural question is: even if the original formula shows a community structure, could it be the case that this structure is quickly destroyed during the search process? We think that the (indirect) exploitation of the community structure is responsible for the success of SAT solvers based on learning and restarting techniques. Thus, the second question is: how is this statement supported?

Let us try to answer the second question. If we do not exploit the community structure, in a pure DPLL schema, the best thing we can do is to assign the variables that occur more often in small clauses. Then, every time we assign $x = true$ we can remove all clauses containing x (because they become satisfied) and remove the literal $\neg x$ from the clauses that contain it, possibly creating unary clauses that force the assignment of a variable. However, if most frequent variables belong to distinct communities, and most clauses are local to a community, we will loose a lot of time deciding about variables that do not contribute to falsify any clause. If we use learning, we can revert this situation. If most of the original clauses are local,[4] since learned clauses are obtained by resolution from them, with high probability the learned clauses will be also local. When the solver learns $x_1 \vee \ldots \vee x_n$ after a conflict, the solver has already decided the assignment for all these variables, among others. I.e. the sequence of decision is something like

$$y_1, \ldots, y_{m_1}, x_1, \cdots, x_{n-1}, y_{m_{n-1}+1}, \ldots, y_{m_n}, x_n$$

Deciding about the y variables has been useless. Moreover, possibly we will repeat this work for the opposite assignment of those variables, that being in other community, have no influence in our conflict. However, now the solver can backtrack to the decision of x_{n-1}, not reconsidering $y_{m_{n-1}+1}, \ldots, y_{m_n}$. Moreover, we can restart the execution, and if we use an activity-based heuristic, next time the solver will try to decide on the variables that have been participating in recent conflicts, i.e. with higher probability on the x variables . We think that, this way, the solver centers its attention on the variables of a single community, or a few communities. Obviously, we could exploit the community structure directly, if previously we had computed a good partition. We can apply learning locally in each community, and re-start the algorithm with a lot of supplementary clauses.

To test the thesis that a considerable part of learning is performed locally inside one or a few communities, we generate formulas with the learned clauses and analyze their modularity. We use the picosat SAT solver [3] (version 846), since it incorporates a conflict directed clause learning algorithm, activity-based heuristics, and restarting strategies.

For the VIG model, we use the original formula to get a partition of the vertices, i.e. of the variables, into communities. Then, we use modularity as a *quality measure* to see how good is the same partition, applied to the graph obtained from the set of learned clauses. Since both graphs (the original formula and the learned clauses) have the same set of vertices (the set of variables), this can be done directly.

[4] A clause is local, in the VIG model, if all its variables belong to the same community, and in the CVIG model, if it is only connected to variables of the same community.

For the CVIG model we must take into account that the graph contains variables and clauses as vertices. Therefore, the procedure is more complicated. We use the original formula to get a partition. We remove from this partition all clauses, leaving the variables. Then, we construct the CVIG graph for the set of learned clauses. The partition classifies the variables of this second graph into communities, but not the clauses. To do this, we assign to each clause the community of variables where it has more of its variables included. In other words, given the labels of the variables we apply a single iteration of the label propagation algorithm to find the labels of the clauses.

We want to see how fast is the community structure degraded along the execution process of a SAT solver. Therefore, we have repeated the experiment for just the first 100 learned clauses and for all the learned clauses. We also want to know the influence of the quality of the learned clauses. Therefore, we also repeat the experiment for all the learned clauses (Table 2), and only using the clauses that participate in the proof of unsatisfiability (Table 3). Notice that Table 3 contains fewer entries than Table 2 because we can only consider unsatisfiable instances. Notice also that picosat is not able to solve all 2010 SAT Race instances, therefore Tables 2 and 3 contain fewer instances than Table 1. The analysis of the tables shows us that the CVIG model gives better results for the original formula and the first 100 learned clauses, but equivalent results if we consider all learned clauses. There are not significant differences if we use all learned clauses, or just the clauses that participate in the refutation. Finally, there is a drop-off in the modularity (in the quality of the original partition) as we incorporate more learned clauses. This means that, if we use explicitly the community structure to improve the efficiency of a SAT solver, to overcome this problem, we would have to recompute the partition after some number of variable assignments to adjust it to the modified formula.

It is worth to remark that, in Table 2, the average IEF, for the VIG and the CVIG models and the first 100 learned clauses, is respectively 0.72 and 0.59.

5. Conclusions

The research community on complex networks has developed techniques of analysis and algorithms that can be used by the SAT community to improve our knowledge about the structure of industrial SAT instances, and, as result, to improve the efficiency of SAT solvers.

In this paper we address the first systematic study of the community structure of SAT instances, finding a clear evidence of such structure in most analyzed instances. We discuss how this structure is already exploited by modern SAT solvers. In fact, some features, like Moskewicz's activity-based heuristics, were already designed thinking on the existence of this kind of structure. Here we go a step further, and propose an algorithm that is able to compute the communities of a SAT instance. It verifies the assumption about the existence of this community structure. The algorithm can also be used directly by SAT solvers to focus their search.

References

[1] E. Amir and S. A. McIlraith. Partition-based logical reasoning for first-order and propositional theories. *Artif. Intell.*, 162(1-2):49–88, 2005.

[2] M. J. Barber. Modularity and community detection in bipartite networks. *Phys. Rev. E*, 76(6):066102, 2007.
[3] A. Biere. Picosat essentials. *JSAT*, 4(2-4):75–97, 2008.
[4] A. Biere and C. Sinz. Decomposing sat problems into connected components. *JSAT*, 2(1-4):201–208, 2006.
[5] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner. On modularity – NP-completeness and beyond. Technical report, Faculty of Informatics, Universität Karlsruhe (TH), Tech. Rep. a 2006-19, 2006.
[6] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70(6):066111, 2004.
[7] N. Eén and N. Sörensson. An extensible sat-solver. In *6th Int. Conf. on Theory and Applications of Satisfiability Testing (SAT'03)*, pages 502–518, 2003.
[8] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.
[9] I. P. Gent, H. H. Hoos, P. Prosser, and T. Walsh. Morphing: Combining structure and randomness. In *AAAI/IAAI*, pages 654–660, 1999.
[10] C. P. Gomes and B. Selman. Problem structure in the presence of perturbations. In *AAAI/IAAI*, pages 221–226, 1997.
[11] C. P. Gomes, B. Selman, and H. A. Kautz. Boosting combinatorial search through randomization. In *Proc. of the 15th Nat. Conf. on Artificial Intelligence (AAAI'98)*, pages 431–437, 1998.
[12] T. Hogg. Refining the phase transition in combinatorial search. *Artif. Intell.*, 81(1-2):127–154, 1996.
[13] M. Järvisalo and I. Niemelä. The effect of structural branching on the efficiency of clause learning sat solving: An experimental study. *J. Algorithms*, 63:90–113, January 2008.
[14] M. W. Moskewicz, C. F. Madigan, Y. Zhao, L. Zhang, and S. Malik. Chaff: Engineering an efficient sat solver. In *DAC*, pages 530–535, 2001.
[15] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Phys. Rev. E*, 69(6):066133, 2004.
[16] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69(2):026113, 2004.
[17] U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E*, 76(3):036106, Sep 2007.
[18] A. Slater. Modelling more realistic sat problems. In *Australian Joint Conference on Artificial Intelligence*, pages 591–602, 2002.
[19] T. Walsh. Search in a small world. In *IJCAI*, pages 1172–1177, 1999.
[20] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998.
[21] R. Williams, C. P. Gomes, and B. Selman. Backdoors to typical case complexity. In *IJCAI*, pages 1173–1178, 2003.

On 2SAT-MaxOnes with Unbalanced Polarity: from Easy Problems to Hard MaxClique Problems[1]

Josep ARGELICH [2], Ramón BÉJAR, Cèsar FERNÁNDEZ and Carles MATEU
Dept. of Computer Science, Universitat de Lleida, Spain

Abstract. We study the complexity of solving a variant of the Random 2SAT-MaxOnes problem, a variant created by introducing an additional parameter p to control the probability of having positive literals in the clauses of the problem instance, and show that its complexity pattern has interesting properties. This parameter p allows us to generate problem instances ranging from exceptionally hard optimization instances, when $p = 0$, that are equivalent to MaxClique instances, to trivial problem instances when $p = 1$. We show that our problem presents an easy-hard-easy complexity pattern, even on the hardest case of the problem ($p = 0$) where instances are always feasible. We show that the hardness peak is mainly due to a sudden increase in the cost of finding the optimal solution and that for $p > 0$ a phase transition from feasible to unfeasible instances appears, but with a lower hardness peak as p approaches to 1. We further investigate the reasons for such a peak, by analyzing the expected number of optimal solutions and expected number of variables set to 1 in optimal solutions.

Keywords.
2SAT, Satisfiability, MaxSAT, Phase Transitions

Introduction

The study of typical-case complexity of decision problems has helped to develop and improve algorithms for SAT and other combinatorial problems, and defining random distributions of problem instances and studying their typical-case complexity has become a very active research area. In the last few years, the interest of using the MaxSAT and partial MaxSAT formalisms for encoding and solving optimization problems has grown significantly [1,2,3], as has the interest in understanding their typical-case complexity [4] and in developing efficient solvers [5]. In particular, complexity transitions for pure optimization problems (problems that always have feasible solutions) have been studied for the asymmetric TSP [6] and MaxSAT [7,8], and phase transition behaviour for problems with not always feasible solutions has been studied, for example, for partial MaxSAT [8] and for a connection sub-graph problem [9].

[1]The research supported by the Spanish CICYT Projects ARINF (TIN2009-14704-C03-01) and TASSAT (TIN2010-20967-C04-03).

[2]Corresponding Author: Department of Computer Science, Universitat de Lleida. Jaume II, 69. 25001, Lleida. SPAIN. E-mail: jargelich@diei.udl.cat

In this work, we study two important aspects of optimization problems. The relation between different complexity patterns and control parameters, and the complexity of finding the optimal solution versus proving its optimality.

To perform the study, we select a variant of the Random 2SAT-MaxOnes problem, where we can control the proportion of positive and negative literals present in the clauses of the formula with a probability parameter p. We call this problem the Random-p 2SAT-MaxOnes problem. This parameter allows us to generate from exceptionally hard problem instances that are always feasible for any ratio of clauses to variables (c/n), when $p = 0$, to problem instances of lower and lower complexity, as p approaches to 1, but that are always NP-hard for any $p < 1$. Even if our problem is NP-hard, it can encode the (hard to approximate) Maximum Clique problem, it is simple enough to allow some analysis of its properties, making it a very attractive problem for complexity analysis.

Our experimental results show an easy-hard-easy complexity pattern for our problem and that the search cost is dominated by the cost of proving the optimality of the solution, as finding an optimal solution is relatively easier. As a possible partial explanation, we analyze the expected number of optimal solutions for a given ratio c/n, and discuss a possible justification which is consistent with previous results about the solution-space geometry of satisfiable Random K-SAT formulas [10,11]. For the not always feasible cases ($0 < p < 1$), we observe an abrupt phase transition from feasible to unfeasible instances, that we analyze generalizing the analysis for the classical Random 2SAT problem. We also observe that the cost of the feasible instances dominates the cost of the unfeasible ones, so that the hardest instances are always located just to the left of the phase transition. We observer that in the peak of hardness, the time required to solve problems decreases as p approaches 1, although our experimentation suggests that the scaling cost at the peak of hardness is exponential for $0 < p < 1$

1. Complexity Patterns for Optimization Problems

A great majority of optimization problems present easy-hard complexity patterns, like the well known MaxSAT problem, so that when adding more clauses the cost of solving the problem increases, although the increase is more abrupt for low ratios of clauses to variables. However, there exist optimization problems where easy-hard-easy patterns have been found. For example, in [6] the authors present a detailed analysis of the hardness of solving the asymmetric TSP, and show that depending on the model of distribution of problem instances, the pattern can be either easy-hard or easy-hard-easy. Finally, in [8] the complexity of different variations of the problems MaxSAT and SAT-MaxOnes are studied and compared.

2. The Random-p 2SAT-MaxOnes Problem

The 2SAT-MaxOnes problem is the problem of finding, among the solutions of a 2SAT instance, the ones that maximize the number of variables set to 1. It is NP-hard, as for example, we can encode the Maximum Clique problem as a 2SAT-MaxOnes instance where all the literals in the clauses are negative [12]. Given the hardness of the Maximum Clique problem, one can expect that at least this special case of 2SAT-MaxOnes (all the

literals in the clauses are negative), will be as hard as Maximum Clique. So, a natural question arises then, that is, what happens with the complexity when positive literals are present in the clauses. The Random-p 2SAT-MaxOnes problem is a variant of Random 2SAT-MaxOnes where the literals in the clauses are generated with positive polarity with probability p. This parameter allows us to generate a wide spectrum of problems, from the hard Maximum Clique instances (when $p = 0$) to trivial problem instances (when $p = 1$).

3. Experimental Results

3.1. Complexity Patterns and Phase Transition

Our main motivation for studying Random-p 2SAT-MaxOnes in the special case where all literals in the clauses are negative (Random-0 2SAT-MaxOnes), is that it seems the most simple case one can consider of the MaxOnes problem that satisfies two interesting properties; on one hand, it always has feasible solutions, and on the other hand, it can be used to encode an exceptionally hard optimization problem, the MaxClique problem, that it is even hard to approximate within $n^{(1-\epsilon)}$ for any $\epsilon > 0$ [13].

So, we begin by investigating the Random-0 2SAT-MaxOnes problem. Given a number of variables (n), and a number of clauses (c), we generate 300 instances and calculate the mean number of backtracks to solve to optimality the instances with WMaxSatz [14] (version 2.5). Figure 1 shows the results for $n = 75$ and $n = 100$, where the horizontal axis shows the ratio c/n of the instances. We observe an easy-hard-easy pattern in the complexity of the instances (in the figures labelled as *Bk*), with the hardest instances located around the ratio 1.6. Although the descent of hardness on the right side is not as abrupt as the one typically observed in decision problems, observe that as the number of variables grows, the descent is steeper. To further investigate the reasons for such a peak, we have measured the search cost for finding the optimal solution (shown in the figure labelled as *BkOpt*). We observe that as the ratio increases to the peak of hardness (≈ 1.6), the cost of finding the optimal solution increases abruptly, and then from that point it decreases, in such a way that from the peak of hardness it is almost a constant fraction of the total cost. [3] We have also computed the mean number of variables set to 0 in the optimal solutions. This information is shown in the plots, labelled as *#unsat* and normalized in such a way that the highest possible number of variables coincides with the highest number of backtracks for the search cost. We observe that the more constrained the problem is, the higher the number of zeros in the optimal solution, and that at the peak of hardness the number of variables set to 0 is about half the number of variables. In a later section, we present a more detailed analysis of the behaviour of the problem and of the expected number of optimal solutions.

For the cases where $p > 0$, a phase transition from feasible to unfeasible instances appears, but the hardest instances are still located at a ratio where only feasible instances exist, that is, in the under-constrained region. This was observed before for the particular cases of Random-$\frac{1}{2}$ 2SAT-MaxOnes and Random-$\frac{1}{2}$ 3SAT-MaxOnes [8]. We observe that from $p = 0.0$ to $p = 0.5$, the ratio c/n for the peak of hardness moves from 1.6 to 0.8

[3] The difference between the total time and the time for finding the optimal solution is the time needed to certify optimality.

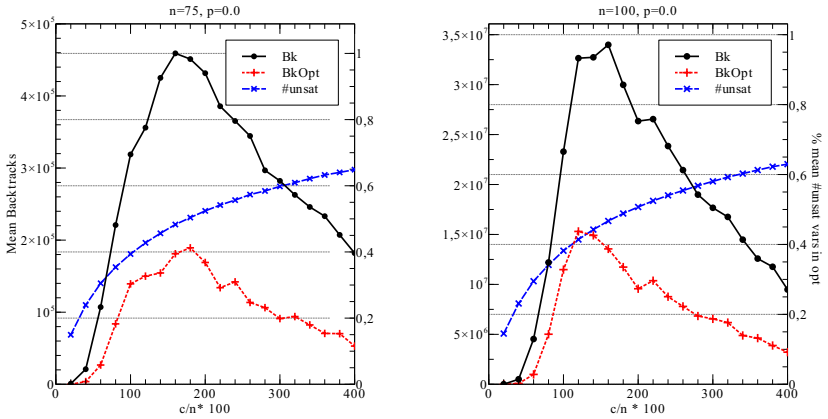


Figure 1. Random-0 2SAT-MaxOnes

(there is actually an abrupt change from $p = 0.0$ to $p = 0.1$, when it changes from 1.6 to 1.0), and then from $p = 0.5$ to $p = 0.9$ it increases again up to, approximately, 1.8. We observe a similar situation with the location of the threshold for the phase transition, this will be analysed in more detail in the last section of the paper. Figure 2 shows the mean number of backtracks, mean number of feasible instances and mean number of negative variables (variables set to 0) in optimal solutions for $n = 100$ and $p = 0.1, 0.5$. The figure also shows the mean number of backtracks for finding the optimal solution, so we can observe that with $p = 0.1$, as with the case $p = 0.0$, the total cost is dominated by the cost of finding the optimal solution, but when $p = 0.5$, that gives instances of much lower complexity, the relative importance of the cost of finding optimal solutions is now not as high as before. Also, at the peak of hardness, the number of negative variables in the optimal solutions decreases as p increases, that is consistent with the fact that the more positive literals in the clauses we have, the more positive variables we can potentially have in optimal solutions. So, the set of optimal solutions will have a lower entropy than for the $p = 0$ case, probably resulting in a less hard solution space geometry than for $p = 0$, a conjecture about the reasons for the hardness peak will be introduced in detail in a following section.

Even if the typical complexity decreases as p increases, as we can observe by comparing figures 1 and 2, it turns out that in the worst case, 2SAT-MaxOnes is NP-hard when there is a linear fraction of clauses with only negative literals. However, this does not mean that typical instances of *Random-p* 2SAT-MaxOnes are equally hard as p increases. Figure 3 shows that the cost of solving the instances at the peak of hardness, for different values of p starting at $p = 0.5$, decreases as p increases, although the scaling seems to be exponential for all cases.

3.2. The Easy-hard-easy Pattern for $p = 0$

Our experimental results show an intriguing easy-hard-easy pattern for Random-0 2SAT-MaxOnes as the ratio of clauses to variables increases, with a peak of hardness located

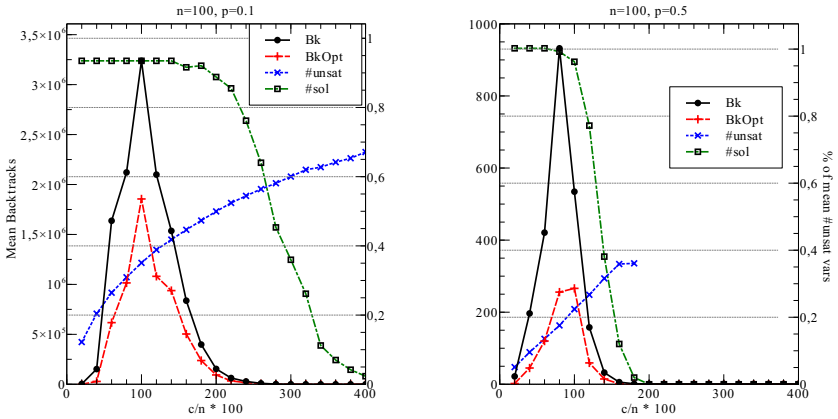


Figure 2. Random-p 2SAT-MaxOnes with $p = 0.1$ and $p = 0.5$

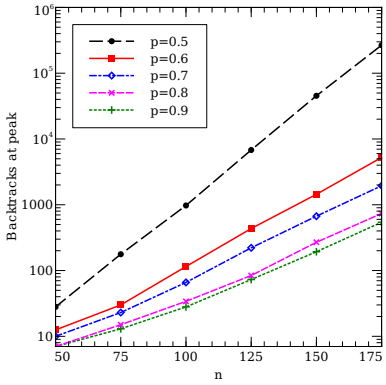


Figure 3. Cost scaling for Random-p 2SAT-MaxOnes from p=0.5 to p=0.9

around the ratio 1.6. We can, informally, justify why the problem should be easy at the extreme values for the ratio, and provide a possible explanation for the peak of hardness.

At low values of c/n, many variables can be safely set to 1 as this will give, with high probability, an optimal solution of the problem. Note that such a probability becomes higher as the value c/n is lower. The higher the number of variables with value 1 in an optimal solution, the easier it will be for any algorithm that prunes the search space to quickly discard other assignments that cannot improve the optimal solution. In the limit, when c/n tends to 0, the unique optimal solution will have the value 1 for all the variables, and this solution will be found, and proven to be optimal, very quickly.

At high values of c/n, setting a variable to 1 will force the algorithm, by just using unit propagation on the clauses, to set to 0 a high number of variables. So, unit propagation will discover the value of many variables (to be 0) by just setting to 1 a few of them. As the objective function forces the algorithm to set to 1 as many variables as possible,

this means that the optimal solutions will have a high number of variables set to 0, but since this is discovered by unit propagation, the algorithm will find an optimal solution quickly. In the limit, when $c/n \to (n-1)/2$, every time the algorithm sets a variable to 1, unit propagation will set the value of all other variables to 0. That will be an optimal solution of the instance, and the algorithm will quickly discover the other possible optimal assignments (the ones that we get if we change the variable set to 1). So, at the two extreme values for the ratio (0 or $n-1$), we have either one optimal solution or a polynomial number of solutions (n), and in both cases they can be discovered and proved to be optimal quickly.

One can think of an intermediate ratio where solutions have about $n/2$ variables set to 1, but it is difficult to discover such $n/2$ positive variables in one solution, so finding such optimal solutions will be difficult. We conjecture that this is actually the case at the peak of hardness: optimal solutions have around $n/2$ positive variables, but finding such optimal solutions is hard because it is possible to set a high number of variables to 1, before discovering whether this partial assignment can really be completed as an optimal solution.

To further check whether our conjecture may be true, we have analyzed the expected number of solutions with i positive variables ($E[N_i]$). We have observed that $E[N_i]$, when plotted for every ratio c/n with i equal to the average number of positive variables in the optimal solutions, has a maximum peak just on the ratio where the cost of solving the instances starts to abruptly increase. We have also observed that the region with the maximum increase in solving cost coincides with the region where the expected number of optimal solutions starts to decrease. These results indicate that in our problem something similar to what happens in Random K-SAT formulas [10,11] could explain the peak of hardness. The results for Random K-SAT indicate that at some point below its satisfiability threshold, the set of satisfying assignments of a K-SAT formula decomposes into an exponential number of well-separated small clusters. A cluster is a set of solutions such that they form a connected component in the graph of solutions where two solutions are adjacent if their Hamming distance is 1. For large K and the density c/n sufficiently close to the threshold, just before solutions disappear, clusters become arbitrary small and far apart, and this is believed to be the main reason for no having polynomial time algorithms for such satisfiable random K-SAT formulas.

Figure 4 shows, for Random-0 2SAT-MaxOnes with $n = 100$, the mean cost of solving the instances (solid plot on the left) and the mentioned value $E[N_i]$ (plot on the right). For the plots of the left, we also show the mean cost for finding the first optimal solution (dashed plot) and the ratio of the total cost to the cost of finding the optimal solution (dotted plot). Observe that just when this last ratio begins to smooth its descent (from $c/n = 0.5$), the solving cost abruptly increases, precisely for $c/n = 0.5$ where the expected number of optimal solutions starts to decrease, after reaching its maximum at $c/n = 0.5$. The peak of hardness ($c/n \approx 1.6$) is located between the maximum and minimum point for the number of solutions shown on the right plot, being so consistent with a point where the number of optimal solutions has decreased enough to separate them far apart into different clusters, but still high enough to support an exponential number of such clusters.

To analyze the expected number of solutions with i positive variables, we have used the expression

Table 1. Number of positive variables in Random-0 2SAT-MaxOnes as measured on experiments and estimated upon Eq. 1

	V=50		V=75		V=100	
c/n	Eq. 1	meas.	Eq. 1	meas.	Eq. 1	meas.
0	50	50	75	75	100	100
40	40	38	61	57	82	76
80	35	32	53	49	71	65
120	31	28	47	43	64	57
160	28	26	43	39	58	52
200	26	24	39	36	53	48

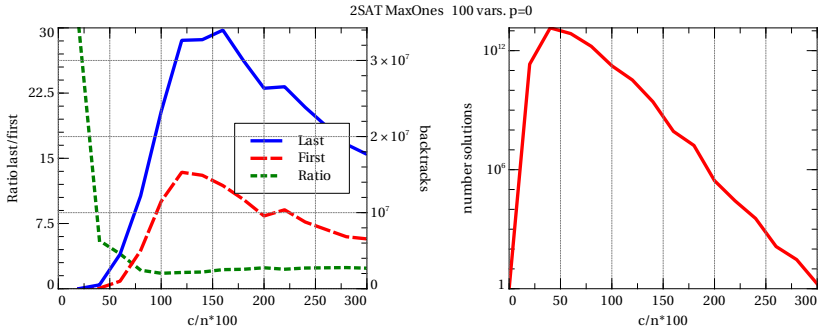


Figure 4. (Left): number of backtracks to find the first optimal solution (first) and to prove its optimality (last). Ratio between both values (ratio). (Right): number of expected optimal solutions according to Eq. 1 for the number of variables set to 1 as measured on Table 1

$$E[N_i] = \binom{n}{i} (1 - A_i)^c \tag{1}$$

where A_i is the probability that a random assignment with i positive variables does not satisfy a Random-p 2SAT clause. That is,

$$A_i = \frac{p^2(n-i)(n-i-1)}{n(n-1)} + \\ + \frac{2p(1-p)i(n-i) + (1-p)^2 i(i-1)}{n(n-1)}$$

So, a possible approximation (actually, experiments show that it is an upper bound) to determine the number of positive variables of a Random-p 2SAT-MaxOnes optimal solution could be the value of i for which Eq. 1 becomes less than 1. Table 1 shows the measured values (obtained from our experiments) for the average number of positive variables in the optimal solutions (mean over 300 instances) and its corresponding upper bound based on the first value of i for which Eq. 1 is below 1, for $p = 0$ and number of variables equal to 50, 75 or 100. It seems that there is actually a sharp threshold for $E[N_i]$, as the value of i in the optimal solutions in our experiments gives a high value for $E[N_i]$, but it quickly drops below 1 just with a few more variables set to 1.

4. The 2SAT-MaxOnes Phase Transition

The satisfiability transition for Random-p 2SAT-MaxOnes is determined by the set of binary clauses contained in a Random-p 2SAT problem. Here, we denote as Random-p 2SAT a typical Random 2SAT problem where the polarity of each literal is chosen positive with probability p and negative with probability $(1-p)$. With the current notation, the classical Random 2SAT problem is Random-$\frac{1}{2}$ 2SAT.

The phase transition for Random 2SAT problems is well established [15], and we know that there is a sharp transition when $\frac{c}{n} = 1$. The analysis is based on the fact that every unsatisfiable formula has a cycle of assignments in normal form (see [15] for the definition). If one is able to compute the mean value of the number of cycles in normal form, by Markov's inequality the unsatisfaction probability of a given formula is upper bounded, resulting these upper bound far more accurate than those based on the first and second moments of the number of solutions, as used for Random K-SAT problems in [16].

If we are able to compute the mean value of the number of cycles in normal form we have, by Markov's inequality, an upper bound on the the unsatisfactibility probability of a given formula. And these upper bound are far more accurate than those based on the first and second moments of the number of solutions, as used for Random K-SAT problems in [16].

Actually, the results for Random K-SAT based on the first and second moment for the number of solutions are more precise as the value of k grows.

Goerdt [15] shows that the mean number of simple cycles of length l is $o\left((c/n)^l\right)$, deriving so the above mentioned sharp threshold. Such calculations can be extended for Random-p 2SAT problems considering that any simple cycle of length l, when expressed as 2SAT clauses, has exactly l clauses, with l positive literals and l negative literals. Now, we have to consider the generation model. In our case, we pick c pairs of different variables among $N = \binom{n}{2}$ at random and we negate each variable with probability $1-p$. If so, the probability of having a given simple cycle (π) of length l is

$$P_\pi = p^l(1-p)^l \frac{\binom{N-l}{c-l}}{\binom{N}{c}}.$$

Considering that a cycle of length l is composed by $v = l/2 - 1$ variables, we can obtain the number of distinct cycles of length l as

$$\mu = n \cdot \binom{n}{v} \cdot \left(v! \cdot 2^{v-1}\right)^2 \cdot \binom{n-1-v}{v}$$

and compute the expected number of cycles of length l, X, as

$$E[X] = \mu \cdot P_\pi$$

Considering that the unsatisfaction probability for an instance and Markov's inequality, we obtain

$$Pr(Unsat) = P(X \geq 1) \leq E[X] \tag{2}$$

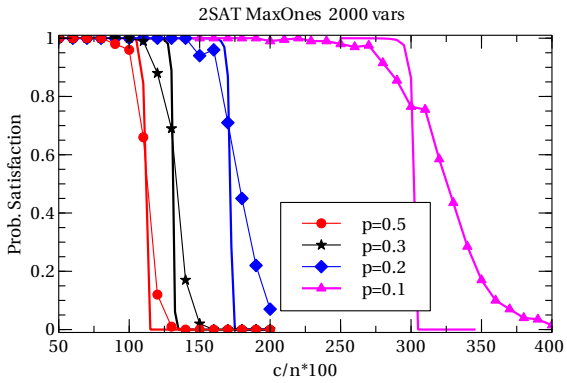


Figure 5. Experimental (solid line with points), numerical (solid line) phase transition for 2SAT-MaxOnes with 2000 variables

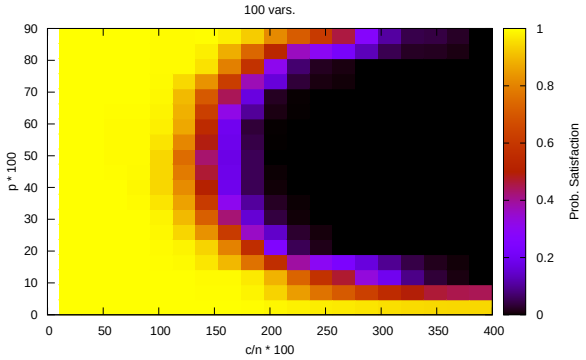


Figure 6. Phase transition for Random-p 2SAT-MaxOnes

Figure 5 shows the curves for the phase transition obtained from our experiments and the curves for the phase transition obtained from Eq. 2 for 2000 variables as a function of p. It is worth noticing that the predicted threshold ratio is very close to the real one, although the prediction is better for values of p near to 0.5.

Figure 6 shows the the curves for the phase transition obtained from our experiments for n=100, and for values for p from 0.0 to 0.9. Observe that the curves are symmetric around the value $p = 0.5$, as it should be, because the satisfiability of a set of 2SAT clauses does not change if we rename positive literals to negative ones, and negative literals to positive ones.

5. Conclusions and further work

In this work we have considered a well known branch and bound style exact algorithm for partial MaxSAT, for solving our variant of the MaxOnes problem, but in the near future we plan to investigate the behaviour of another class of optimization algorithms that is proving very successful for industrial applications of partial MaxSAT: the unsatisfiability cores based algorithms [17,18,19].

References

[1] J. Argelich, I. Lynce, and J. P. M. Silva, "On solving boolean multilevel optimization problems," in *Proc. of IJCAI 2009*, pp. 393–398, 2009.
[2] Y. Chen, S. Safarpour, A. G. Veneris, and J. P. M. Silva, "Spatial and temporal design debug using partial MaxSAT," in *Proc. of 19th ACM Great Lakes Symposium on VLSI*, pp. 345–350, 2009.
[3] A. Graça, I. Lynce, J. Marques-Silva, and A. L. Oliveira, "Haplotype inference combining pedigrees and unrelated individuals," in *WCB09*, September 2009.
[4] R. Béjar, A. Cabiscol, F. Manyà, and J. Planes, "Generating hard instances for MaxSAT," in *Proc. of ISMVL'09*, pp. 191–195, 2009.
[5] J. Argelich, C. M. Li, F. Manyà, and J. Planes, "The first and second MaxSAT evaluations," *JSAT*, vol. 4, no. 2-4, pp. 251–278, 2008.
[6] W. Zhang and R. E. Korf, "A study of complexity transitions on the asymmetric traveling salesman problem," *Artificial Intelligence*, vol. 81, no. 1-2, pp. 223 – 239, 1996.
[7] W. Zhang, "Phase transitions and backbones of 3-SAT and maximum 3-SAT," in *Proc. of CP 2001*, pp. 153–167, Springer, 2001.
[8] J. K. Slaney and T. Walsh, "Phase transition behavior: from decision to optimization," in *Proc. of SAT2002*, 2002.
[9] J. Conrad, C. P. Gomes, W. J. van Hoeve, A. Sabharwal, and J. Suter, "Connections in networks: Hardness of feasibility versus optimality," in *Proc. of CPAIOR2007*, pp. 16–28, 2007.
[10] D. Achlioptas and F. Ricci-Tersenghi, "On the solution-space geometry of random constraint satisfaction problems," in *Proc. of STOC'06*, pp. 130–139, 2006.
[11] D. Achlioptas, "Solution clustering in random satisfiability," *Eur. Phys. J. B*, vol. 64, pp. 395–402, 2008.
[12] B. Cha, K. Iwama, Y. Kambayashi, and S. Miyazaki, "Local search algorithms for partial MaxSAT," in *AAAI'97*, pp. 263–268, 1997.
[13] J. Håstad, "Clique is hard to approximate within $n^{(1-\epsilon)}$," in *FOCS'06*, pp. 627–636, 1996.
[14] C. M. Li, F. Manyà, N. O. Mohamedou, and J. Planes, "Exploiting cycle structures in MaxSAT," in *Proc. of SAT 2009*, pp. 467–480, 2009.
[15] A. Goerdt, "A threshold for unsatisfiability," *J. Comput. Syst. Sci.*, vol. 53, no. 3, pp. 469–486, 2006.
[16] D. Achlioptas and Y. Peres, "The threshold for random $k-$SAT is $2^k(ln2 - o(k))$," in *Proc. of STOC 2003*, (New York, NY, USA), pp. 223–231, ACM, 2003.
[17] C. Ansótegui, M. L. Bonet, and J. Levy, "Solving (weighted) partial MaxSAT through satisfiability testing," in *Proc. of SAT 2009*, pp. 427–440, 2009.
[18] V. M. Manquinho, J. P. M. Silva, and J. Planes, "Algorithms for weighted boolean optimization," in *Proc. of SAT 2009*, pp. 495–508, 2009.
[19] Z. Fu and S. Malik, "On solving the partial MaxSAT problem," in *Proc. of SAT 2006*, pp. 252–265, 2006.

Experimenting with the Instances of the MaxSAT Evaluation

Josep ARGELICH [a], Chu Min LI [b], Felip MANYÀ [c] and Jordi PLANES [a]
[a] *Universitat de Lleida, Lleida, Spain*
[b] *Université de Picardie Jules Verne, Amiens, France*
[c] *Artificial Intelligence Research Institute (IIIA-CSIC), Bellaterra, Spain*

Abstract. We report on a number of experiments on the instances of the MaxSAT Evaluation with the aim of gaining new insights into their computational hardness, answering some questions that have been asked to us as organizers, and evaluating how appropriate are the current settings of parameters such as timeout and available RAM memory. The lessons we have learned from the analysis of the empirical results suggest to introduce several modifications in forthcoming evaluations.

1. Introduction

The MaxSAT Evaluation[5][1] is an affiliated event of the International Conference on Theory and Applications of Satisfiability Testing that is held every year since 2006, and is devoted to empirically evaluate exact MaxSAT algorithms solving any of the following problems: MaxSAT, Weighted MaxSAT (WMaxSAT), Partial MaxSAT (PMaxSAT), and Weighted Partial MaxSAT (WPMaxSAT).

The objective of the present paper is to analyze the instances of the 2010 MaxSAT Evaluation (MaxSAT-2010) in order to gain new insights into their computational hardness, answer some questions that have been asked to us as organizers, and evaluate how appropriate are the current settings of parameters such as timeout and available RAM memory. To this end, we have designed a number of experiments, including a comparison of recent MaxSAT solvers with older MaxSAT solvers, a comparison of the performance of solvers when changing the current timeout and memory settings, a comparison of the ranking of solvers when performance is computed taking into account the total number of solved instances versus when it is computed taking into account the percentage of solved instances in each individual instance set, and an analysis of the performance profiles based on parameters such as the size of unsatisfiability cores, and the number of unsatisfied clauses. From the lessons we have learned from the analysis of the empirical results, in the conclusions section we suggest to introduce several modifications in forthcoming evaluations.

The paper is structured as follows. In Section 2, we define basic MaxSAT concepts. In Section 3, we present an overview of the instances of the 2010 MaxSAT Evaluation. In Section 4, we report on the empirical investigation. In Section 5, we present some concluding remarks and suggest modifications in forthcoming evaluations.

[1] http://www.maxsat.udl.cat/

2. Preliminaries

A literal is a propositional variable or a negated propositional variable. A clause is a disjunction of literals. A weighted clause is a pair (C, ω), where C is a clause and ω, its weight, is a natural number or infinity. A clause is hard if its weight is infinity, otherwise it is soft. A WPMaxSAT instance is a multiset of weighted clauses $\phi = \{(C_1, w_1), \ldots, (C_m, w_m), (C_{m+1}, \infty), \ldots, (C_{m+m'}, \infty)\}$, where the first m clauses are soft and the last m' clauses are hard. The cost of a truth assignment I for ϕ is the sum of the weights of the clauses falsified by I.

The WPMaxSAT problem for an instance ϕ consists in finding a truth assignment with minimum cost that satisfies all the hard clauses (i.e, an optimal assignment). The WMaxSAT problem is the WPMaxSAT problem when there are no hard clauses. The PMaxSAT problem is the WPMaxSAT problem when all soft clauses have weight 1. The MaxSAT problem is the PMaxSAT problem when there are no hard clauses. The SAT problem is the PMaxSAT problem when there are no soft clauses.

3. Overview of the Instances

There were 2.675 instances in MaxSAT-2010: 544 MaxSAT instances, 349 WMaxSAT instances, 1.122 PMaxSAT instances, and 660 WPMaxSAT instances.[2] For each problem, instances are assigned to one of the following three categories: random, crafted and industrial.

In the MaxSAT category, there are 300 random instances, 167 crafted instances, and 77 industrial instances. Random instances contain Max-2SAT and Max-3SAT instances; and Max-4SAT and Max-5SAT instances with high girth [1] . Crafted instances contain MaxCut and bipartite MaxCut instances [12]. Industrial instances contain circuit debugging instances [9].

In the WMaxSAT category, there are 200 random instances and 149 crafted instances. Random instances contain WMax-2SAT and WMax-3SAT instances. Crafted instances contain CSP [21], Weighted Max-Cut [12] and Ramsey [12] instances.

In the PMaxSAT category, there are 240 random instances, 385 crafted instances, and 497 industrial instances. Random instances contain Min-2SAT and Min-3SAT instances [18], and PMax-2SAT and PMax-3SAT instances. Crafted instances contain CSP [4,21], Job Shop, Max-Clique and Max-One instances [12], as well as instances from the 2006 PB Evaluation. Industrial instances contain circuit trace compaction [9], binate covering, pseudo-Boolean, haplotype assembly and protein alignment [12] instances.

In the WPMaxSAT category, there are 150 random instances, 378 crafted instances, and 132 industrial instances. Random instances contain WPMax-2SAT and WPMax-3SAT instances. Crafted instances contain Bayesian networks from the Probabilistic Inference Evaluation, and combinatorial auctions [12], pseudo-Boolean [12], and Weighted CSPs [4,12] instances. Industrial instances in WPMaxSAT contain software upgradeability [6], and timetabling instances.[3]

[2] All the instances are publicly available at http://www.maxsat.udl.cat/10/benchmarks/

[3] http://www.lsi.upc.edu/~rasin/timetabling.html

Instance set	#	MiniMS	PMS	IncWMSz	QMaxSat	akms_ls	Toolbar
JobShop	4	178(2)	0(0)	0(0)	**81(4)**	0(0)	0(0)
MC/RANDOM	96	**2.41(96)**	68(96)	3.48(96)	103(76)	9.46(96)	11(96)
MC/STRUC	62	**81(36)**	172(27)	152(36)	130(23)	146(33)	216(26)
MO/3SAT	80	1.29(80)	4(80)	**0.43(80)**	298(72)	1.59(80)	108(80)
MO/STRUC	60	**30(60)**	133(36)	193(54)	9(59)	384(35)	224(43)
PSEUDO/miplib	4	3.08(2)	0.14(2)	0.04(2)	**3.23(4)**	0.29(2)	0.75(2)
frb	25	498(5)	0(0)	376(5)	**246(21)**	1163(5)	1627(1)
min-enc/kbtree	54	203(22)	**188(46)**	380(8)	130(16)	126(19)	2.72(6)
Total	385	**303**	287	281	275	270	254

Table 1. Partial Max-SAT Crafted. Current solvers: IncWMaxSatz (IncWMSz), QMaxSat, and akmaxsat_ls (akms_ls). Past solvers: MiniMaxsat (MiniMS), PMS, and Toolbar.

4. Experiments

Experiments were performed on a cluster with 160 2 GHz AMD Opteron 248 Processors with 1 GB of RAM memory. The solved instances belong to MaxSAT-2010. A total of 17 exact MaxSAT solvers participated in MaxSAT-2010[4]. Solvers can be classified into three main types: branch and bound (B&B) solvers, satisfiability-based (sat-based) and unsatisfiability-based (unsat-based) solvers. In the first type, we find 10 solvers: akmaxsat (akms), akmaxsat_ls (akms_ls) [14], IncMaxSatz, IncWMaxSatz (IncWMSz) [19], Maxsat_Power, LS_Power, WMaxsat_Power (WMS_P) , LSW_Power (LSW_P), WMaxSatz-2009 (WMSz09), and WMaxSatz+ (WMSz+) [17,16]. In the second type, we find 2 solvers: SAT4J-Maxsat (SAT4J) [15], and QMaxSAT (QMS). In the third type, we find 5 solvers: WPM1 [2], PM2, WPM2 [3], wbo 1.4a, and wbo 1.4b [20]. Remind that B&B solvers implement the branch and bound schema and compute lower bounds using resolution techniques. Sat-based solvers solve a sequence of satisfiable instances, where each instance is obtained from the previous one by adding cardinality constraints, and stop when an unsatisfiable instance is found. On the other hand, unsat-based solvers solve a sequence of unsatisfiable instances, and stop when a satisfiable instance is found. Moreover, each time an instance is declared to be unsatisfiable, the detected unsatisfiable core is relaxed by adding a cardinality constraint.

4.1. Experiment 1: Historical Evolution

There have been five editions of the evaluation, and it makes sense to compare how fast are the best solvers of the MaxSAT-2010 compared with the best solvers that participated in previous evaluations but have not been submitted to MaxSAT-2010. So, we compared the 3 best current solvers with the 3 best past solvers on the PMaxSAT instances of the random, crafted and industrial categories. The results for crafted instances are shown in Table 1. The other results are not shown due to lack of space.

Recent solvers are, in general, superior to older solvers although there are some surprising results. The best current solver on random instances (IncWMaxSatz) solves 13,3% instances more than the best past solver (the B&B solver WMaxSatz_icss [10]), on industrial instances (the sat-based solver QMaxSat) solves 14,7% instances more than the

[4] http://www.maxsat.udl.cat/10/solvers/

Solver	# solved	ranking	Solver	# solved	ranking
QMaxSat	294 (+19)	1 (2)	PM2	254 (+22)	8 (7)
IncWMaxSatz	289 (+8)	2 (1)	wbo-1.4b-wcnf	207 (+0)	9 (9)
akmaxsat_ls	281 (+11)	3 (3)	WPM1	69 (+0)	10 (10)
akmaxsat	278 (+8)	4 (4)	WMaxsat_Power	60 (+0)	11 (11)
WMaxSatz-2009	267 (+11)	5 (6)	LSW_Power	60 (+9)	12 (13)
WMaxSatz+	267 (+11)	6 (5)	wbo-1.4a-wcnf	59 (+0)	13 (12)
SAT4J-MAXSAT-2.2.0	258 (+28)	7 (8)	-	-	-

Table 2. PMaxSAT Crafted: timeout of 1800 seconds vs. timeout of 7200 seconds.

best past solver (Chaff_BS [11]), but on crafted instances solves 5,7% instances less than the best past solver (the B&B solver MiniMaxSat [13]) and 1,6% less than the second best past solver (the B&B solver PMS [7]). So, it is important to pay special attention to crafted instances when devising new solving techniques for B&B solvers. On the other hand, we observe that there is still room for improvement because a considerable number of instances are beyond the reach of existing solvers, and the percentages of newly solved instances are probably not so impressive as claimed in some papers. The statistics table[5] of MaxSAT-2010 shows the number of instances solved by at least one solver. We can compare it with the total number of instances in each category, and see the number of instances that are not solved by any participanting solver.

4.2. Experiment 2: Analysis of the Timeout

The second experiment evaluates the impact of setting a timeout of 7200 seconds instead of the timeout of previous evaluations (1800 seconds). The idea is to find out if it is necessary to change the current timeout because it introduces a bias in favor of some solvers. To this end, we solved the PMaxSAT instances of the crafted and industrial category using a timeout of 7200 seconds. We also solved the instances of the random category but are not shown because there are no differences regarding ranking positions. Tables 2 and 3 show the obtained results: the first column corresponds to the name of the solver; the second column corresponds to the number of solved instances with the new timeout and, in brackets, the number of instances that are now solved but were not solved within 1800 seconds; and the third column corresponds to the position of the solver in the ranking with 7200 seconds and, in brackects, the position with 1800 seconds, where ties are broken taken into account the total time needed to solve the instances. We observe that the current timeout does not introduce a significant bias in the sense that the worst variation in the ranking is of at most two positions. Nevertheless, in some cases, we observe a remarkable number of extra instances solved.

4.3. Experiment 3: Analysis of RAM Memory

It is well-known that variations in the amount of available RAM memory may produce quite different performance profiles. So, we decided to evaluate the impact of setting 1GB of RAM instead of 512MB. The cluster used in the evaluation has 2 processors per

[5] http://www.maxsat.udl.cat/10/benchmarks/stats.txt

Solver	# solved	ranking	Solver	# solved	ranking
PM2	373 (+48)	1 (2)	WMaxSatz+	165 (+44)	8 (8)
QMaxSat	349 (+6)	2 (1)	WMaxSatz-2009	164 (+45)	9 (9)
wbo-1.4b-wcnf	298 (+0)	3 (3)	akmaxsat	76 (+8)	10 (10)
SAT4J-MAXSAT-2.2.0	288 (+16)	4 (4)	akmaxsat_ls	66 (+7)	11 (11)
IncWMaxSatz	192 (+48)	5 (7)	WMaxsat_Power	19 (+1)	12 (12)
WPM1	179 (+1)	6 (5)	LSW_Power	17 (+4)	13 (13)
wbo-1.4a-wcnf	165 (+0)	7 (6)	-	-	-

Table 3. PMaxSAT Industrial: timeout of 1800 seconds vs. timeout of 7200 seconds.

Instance set	#	IncWMSz	QMS	akms_ls	akms	WMSz+	WMSz09	WMS_P
JS	4	0(0+0)	**77(4+0)**	0(0+0)	0(0+0)	0(0+0)	0(0+0)	0(0+0)
MC/RAN	96	**3(96+0)**	103(76+0)	9(96+0)	9(96+0)	71(82+0)	71(82+0)	73(32+50)
MC/STR	62	**165(36+0)**	140(23+0)	150(33+0)	179(34+0)	178(18+6)	178(18+6)	185(7+17)
MO/3SAT	80	**0.4(80+0)**	290(72+0)	1(80+0)	2.05(80+0)	183(77+0)	183(77+0)	186(1+76)
MO/STR	60	193(54+0)	7(59+0)	384(35+0)	382(35+0)	153(58+0)	153(58+0)	153(13+45)
PS/miplib	4	0.1(2+0)	**3(4+0)**	0.3(2+0)	0.2(2+0)	0.1(2+0)	0.1(2+0)	0.1(2+0)
frb	25	376(5+0)	**268(21+0)**	1158(5+0)	1043(5+0)	0(0+0)	0(0+0)	0(0+0)
min/kbtree	54	382(8+0)	133(16+0)	**125(19+0)**	141(18+0)	245(19+0)	248(19+0)	256(5+14)
Total	385	**281+0**	275+0	270 +0	270+0	256+6	256+6	60+202

Table 4. PMaxSAT crafted: 1GB of RAM memory. Mean time in seconds.

Instance set	#	IncWSz	akms_ls	akms	WMS_P	WMSz09	LSW_P
AUC/PATHS	88	**8(74+14)**	22(88+0)	22(88+0)	199(6+65)	192(71+0)	190(6+64)
AUC/SCHED	84	114(78+4)	267(75+0)	230(73+0)	65(19+64)	63(83+0)	76(14+68)
PSEUDO/miplib	12	28(3+0)	3.58(2+0)	2.19(2+0)	273(1+2)	258(3+0)	271(1+2)
WCSP/SPOT5/dir	21	5(4+0)	1.19(4+0)	0.55(4+0)	11(1+1)	11(2+0)	11(1+1)
WCSP/SPOT5/log	21	0.62(4+0)	1.44(4+0)	1.28(4+0)	15(1+1)	15(2+0)	15(1+1)
minenc/planning	56	93(38+0)	106(39+0)	90(39+0)	103(12+39)	207(45+5)	110(7+44)
minenc/warehouses	18	3.10(12+0)	0.18(1+0)	0.16(1+0)	0.30(1+0)	0.31(1+0)	0.30(1+0)
random-net	78	1178(1+0)	427(17+0)	429(17+0)	0(0+0)	0(0+0)	0(0+0)
Total	378	**214+18**	230+0	228+0	41+172	207+5	31+180

Table 5. WPMaxSAT crafted: 1GB of RAM memory. Mean time in seconds.

node, and they share 1 GB of RAM. In this experiment, we just use one processor per node; in this way, the available memory is doubled. The timeout was set to 1800 seconds.

Table 4 and Table 5 compare the behavior of solvers using 1GB and 512MB of RAM memory on the crafted instances of PMaxSAT and WPMaxSAT, respectively. The first column contains the name of the instances, and the rest of columns the results obtained for the solver whose name is in the first line. For each solver and set of instances, we show the mean time of the solved instances and, in brackets, the number of instances solved with 512MB followed by the number of extra instances solved with 1GB. The Tables only show the results for the best solvers.

We do not observe, in general, significant differences due to the use of 1GB in the number of solved instances. Looking at the results of the webpage of the MaxSAT-2010,

we also do not observe significant differences in the mean time needed to solve the instances. Nevertheless, there is a surprising results on two B&B solvers: WMaxSat_Power and LS_Power. These solvers combine MaxSatz with local search solvers, and we see that the current results are very close to the results obtained with WMaxSatz+ and WMaxSatz09. So, it seems that the bad behavior with 512MB is due to the way these solvers manage dynamic memory, but not to the solving techniques they implement.

We have also performed the same experiment on the WPMaxSAT instances of the industrial category, obtaining very few differences: wbo-1.4a-wcnf solves 1 instances more, WPM1 solves 3 instances more and WPM2 solves 4 instances more. The ranking of solvers is identical to the ranking for 512MB.

We do not have access to a cluster with more RAM memory, but it would be interesting to prove if SAT-based MaxSAT solvers may have significant gains by using 4GB of memory. The use of more memory does not seem to be relevant for the existing B&B solvers (except for WMaxSat_Power and LS_Power). It would probably be useful for B&B solvers using more sophisticated resolution-based inference rules.

4.4. Experiment 4: Size of Instance Sets

The sets of instances in each category have different size, and we rank solvers by the total number of solved instances. This may bias the results in that there may be sets of instances with a large number of instances and sets with just a few instances. So, one solver solving a lot of instances of a big set and no instances from smaller sets can be ranked higher than a solver solving a substantial number of instances of every set. Therefore, for ranking solvers by their ability to solve instances from different sets, we propose to normalize the results taking into account the number of instances in each set. To this end, we analyzed the situation in all the categories of PMaxSAT and WPMaxSAT. We ignore random instances because all the instance sets have the same size.

Table 6 contains the results of the best solvers for the crafted instances of PMaxSAT, giving the percentage of solved instances instead of the number of solved instances, while Table 7 ranks the solvers according to the total number of solved instances (second column) and the percentage of solved instances (third column). We can see that the results are considerably different with gaps up to four positions. Tables 8 is like Table 7 but for PMaxSAT industrial (left table) and WPMaxSAT crafted (right table). We observe that the rankings are different if the criteria used is the percentage of solved instances. We do not display results for WPMaxSAT industrial because the ranking is identical; in this case there are only two instance sets.

4.5. Experiment 5: Some Parameters of Instances

We have analyzed several parameters of the MaxSAT-2010 instance sets. The results for MaxSAT and WMaxSAT are shown in Tables 9 and 10, respectively. The results for PMaxSAT and WPMaxSAT are not shown due to lack of space. The first column is the category, the second column is the instance set name, the third column is the number of instances, the fourth column is the median number of variables, the fifth column is the median number of clauses, the sixth column is the time spent by solver IncMaxSatz, the seventh column is the mean size of the first core found using picosat [8] within 1800 seconds (when a core is not found for at least one instance, we put in brackets

Instance set	QMS	PM2	SAT4J	IncWMSz	wbo1.4b	akms_ls	akms
JobShop	100%	75%	100%	0%	75%	0%	0%
MAXCLIQUE/RAN	79%	62%	71%	100%	63%	100%	100%
MAXCLIQUE/STR	37%	25%	27%	58%	25%	53%	54%
MAXONE/3SAT	90%	100%	77%	100%	68%	100%	100%
MAXONE/STRUC	98%	65 %	100%	90%	96%	58%	58%
PSEUDO/miplib	100%	100%	100%	50%	75%	50%	50%
frb	84%	52%	0%	20%	0%	20%	20%
min-enc/kbtree	29%	31%	25%	14%	20%	35	33%
Mean (100%)	77%	64%	63%	54%	53%	52%	52%

Table 6. PMaxSAT crafted: Percentages of solved instances per set.

Solver	#solved	%solved	Solver	#solved	%solved
IncWMaxSatz	1	4	SAT4J-MAXSAT-2.2.0	8	3
QMaxSat	2	1	wbo-1.4b-wcnf	9	5
akmaxsat_ls	3	6	WPM1	10	10
akmaxsat	4	7	WMaxsat_Power	11	12
WMaxSatz+	5	8	wbo-1.4a-wcnf	12	11
WMaxSatz-2009	6	9	LSW_Power	13	13
PM2	7	2	-	-	-

Table 7. PMaxSAT crafted: Ranking comparison (# solved instances vs. % solved instances)

the number of instances to which picosat found a core), the eighth column is the mean value of the solutions, the ninth column is the core size multiplied by the solution, and the tenth column is the time spent by the solver PM2/WPM2. We selected IncMaxSatz, and PM2/WPM2 because they are competitive and are representative of B&B and unsat-based solvers, respectively.

A rough remark one can make from the results is that B&B solvers perform better for random instances, while unsat-based and sat-based solvers perform better for industrial instances, and all of them have comparable performance on crafted instances. However, if we look at the results more closely, we can make the two following remarks:

(1) The performance of B&B solvers depends on the size of the problem rather than on the category (random, crafted, or industrial). In fact, with few exceptions, B&B solvers perform better on random and some crafted instances just because they have small size. Even for relatively small industrial instances such as bcp-syn (partial industrial, median #vars=467 and median #clauses=645), B&B solvers may compete with most sat-based and unsat-based solvers. On the other hand, B&B solvers perform poorly for very large instances.

(2) When the instance size is very large, e.g. over 10000 variables and 20000 clauses, sat-based and unsat-based solvers clearly perform better than B&B solvers for the instances used in the evaluation. For instances of smaller size, the relative performance of the two types of solvers seems to depend on the size and the number of the unsatisfiable cores contained in these instances. It is clear, looking at Tables 9–10, that B&B solvers perform generally better than sat-based and unsat-based solvers for MaxSAT and

Solver	#solved	%solved
QMaxSat	1	2
PM2	2	1
wbo-1.4b-wcnf	3	3
SAT4J-MAXSAT-2.2.0	4	4
WPM1	5	5
wbo-1.4a-wcnf	6	6
IncWMaxSatz	7	7
WMaxSatz+	8	8
WMaxSatz-2009	9	9
akmaxsat	10	10
akmaxsat_ls	11	11
WMaxsat_Power	12	12
LSW_Power	13	13

Solver	#solved	%solved
akmaxsat_ls	1	2
akmaxsat	2	3
IncWMaxSatz	3	1
WMaxSatz+	4	4
WMaxSatz-2009	5	5
SAT4J-MAXSAT-2.2.0	6	6
wbo-1.4b-wcnf	7	7
wbo-1.4a-wcnf	8	10
WPM1	9	8
WPM2	10	9
WMaxsat_Power	11	1
LSW_Power	12	12

Table 8. PMaxSAT industrial (left table) and WPMaxSAT crafted (right table): Ranking comparison (# solved instances vs. % solved instances)

cat	instances	#ins.	#vars	#clauses	IncMaxSatz	core size	sol.	est. size	PM2
ran	highgirth/4SAT/	50	125	1125	838.74(23)	-(0)	2	-	272.94(1)
ran	max2sat/120v/	50	120	1400	425.86(45)	12.74	197	2510	0.00(0)
ran	max2sat/140v/	50	140	1400	706.25(27)	13.76	185	2546	0.00(0)
ran	max3sat/70v/	50	70	1000	396.41(49)	67.68	47	3181	0.00(0)
ran	max3sat/80v/	50	80	800	283.94(49)	97.22	27	2625	0.00(0)
cra	Maxcut/DimMod/	62	40	1118	80.85(52)	10.67	192	2049	222.98(10)
cra	Maxcut/spinglass/	5	125	750	28.90(3)	6.4	45	288	3.77(2)
cra	bipart/maxcut-0.7/	50	140	1260	431.64(50)	14.14	166	2347	0.00(0)
cra	bipart/maxcut-0.8/	50	140	1258	310.69(50)	13.84	166	2297	0.00(0)
ind	CircuitDebug/	9	740161	697003	0.00(0)	60.25(4)	5	301	24.16(5)
ind	SeanSafarpour/	68	299452	962946	0.00(0)	354.54(24)	35	12390	111.58(28)

Table 9. MaxSAT: Parameters of instances

cat	instances	#ins.	#vars	#clauses	IncMaxSatz	core size	sol.	est. size	WPM2
ran	wm2sat/100v/	40	100	1371	373.23(35)	11.95	222	2653	0.00(0)
ran	wm2sat/120v/	40	120	1400	444.63(34)	12.75	212	2703	0.00(0)
ran	wm2sat/140v/	40	140	1400	745.99(27)	14.15	201	2844	0.00(0)
ran	wm3sat/hi/	40	70	850	177.47(40)	75.27	41	3086	0.00(0)
ran	wm3sat/lo/	40	70	450	2.05(40)	144.27	11	1587	63.08(18)
cra	Ramsey/	48	72	1144	7.49(36)	2106.08(45)	1	2106	10.26(35)
cra	WMcut/DimMod/	62	40	1118	52.52(55)	10.67	218	2326	0.22(3)
cra	WMcut/spinglass/	5	125	750	34.58(4)	17.8	80	1424	0.00(0)
cra	frb/	34	325	10934	134.15(14)	3	129	387	126.27(15)

Table 10. WMaxSAT: Parameters of instances

WMaxSAT crafted instances. However, when the core size is small (6.4 for Maxcut spinglass and 3 for frb) and the number of cores is small (45 for maxcut/spinglass and 129 for frb), the difference between B&B and the other solvers is not so big, one unsat-based solver, WPM2, is even slightly better than the B&B solvers for frb instances.

For partial instances, the hard clauses complicate the situation. When the hard clauses are very simple like in Maxclique and Auctions, where all hard clauses are binary with negative literals, B&B solvers are clearly better, even if the instances are very large, since clause learning in the underlying SAT solver of sat-based and unsat-based MaxSAT solvers does not play an important role for these binary clauses. For other partial crafted instances, the core size and the number of cores seem to determine the relative performance of the solvers. For example, unsat-based solvers are better for the two WCSP sets because the core size is only 3 and their numbers are respectively 93 and 111, B&B solvers are better for the kbtree instances, because the core size is 140.48, the performance of the two types of solvers are comparable for miplib instances because these are very large (i.e. more than 20000 clauses), and their unsatisfiable cores are also very large (i.e. containing 292.16 clauses on average).

Of course, there may be many other factors in the hardness of a NP-hard problem such as MaxSAT. However, we believe that the problem size, the first size of the unsatisfiable core, and the number of unsatisfiability cores can give very useful indications when selecting a MaxSAT solver to solve a given MaxSAT instance: when the instance has fewer than, e.g. 5000 clauses, use a B&B solver; otherwise, search for a unsatisfiable core of the instance, if the core contains more than, e.g. 10 clauses, again use a B&B solver, if the hard clauses of the instances are of very simple form (e.g. binary clauses with negative literals), always use a B&B solver. In all other cases, use a sat-based or unsat-based solver. Regarding sat-based solvers, which are good on large size instances, it seems to be decisive the quality of the first upper bound.

5. Conclusions

We have performed a number of experiments which allow us to have a clearer picture of the instances of the MaxSAT Evaluation. Some of the gained insights suggest to introduce several modifications in forthcoming evaluations:

Experiment 1 suggests that we should consider the best previous solver for each problem and category until it is beaten by new solvers. On the other hand, taking into account the number of unsolved instances not yet solved by any participating solver, we should report in the results of the evaluation the number of solved instances that have been solved for the first time.

Experiment 2 suggests that the current timeout is adequate for the evaluation. The introduction of a higher timeout could complicate the development of the evaluation without introducing significant differences in the conclusions of the evaluation.

Experiment 3 suggests that the fact of doubling the available RAM memory does not lead to remarkable differences in performance. However, it is interesting to double the memory from time to time in order to detect anomalous situations such as the one we detected for Maxsat_Power and LSW_Power. On the other hand, it would be interesting to perform the evaluation with a cluster allowing 4GB or more of RAM memory to every solver, but this is beyond the reach of the organizers for the time being.

Experiment 4 suggests that it is difficult to set a minimum number of instances per set because there can be sets with very few real-world instances, but it makes sense to limit the maximum number of instances per set. Therefore, our proposal is to set a maximum of 100 instances per set, and present the results using both the ranking based on total number of solved instances and the ranking based on percentage of solved instances.

Experiment 5 suggests that there are a number of parameters that may help predict the type of solver that is more suitable for a set of instances. Parameters that could be reported in the evaluation include median number of variables/clauses, number of unsatisfied clauses, and core size.

References

[1] C. Ansótegui, R. Béjar, C. Fernández, and C. Mateu. From high girth graphs to hard instances. In *CP-2008, Sydney, Australia*, pages 298–312. Springer LNCS 5202, 2008.

[2] C. Ansótegui, M. L. Bonet, and J. Levy. Solving (weighted) partial MaxSAT through satisfiability testing. In *SAT-2009, Swansea, UK*, pages 427–440. Springer LNCS 5584, 2009.

[3] C. Ansótegui, M. L. Bonet, and J. Levy. A new algorithm for weighted partial MaxSAT. In *AAAI-2008, Atlanta, USA*, pages 3–8, 2010.

[4] J. Argelich, A. Cabiscol, I. Lynce, and F. Manyà. New insights into encodings from MaxCSP into partial MaxSAT. In *ISMVL-2010, Barcelona, Spain*, pages 46–52. IEEE CS Press, 2010.

[5] J. Argelich, C. M. Li, F. Manyà, and J. Planes. The first and second Max-SAT evaluations. *Journal on Satisfiability, Boolean Modeling and Computation*, 4:251–278, 2008.

[6] J. Argelich, I. Lynce, and J. Marques-Silva. On solving Boolean multilevel optimization problems. In *IJCAI-2009, Pasadena, California, USA*, pages 393–398, 2009.

[7] J. Argelich and F. Manyà. Partial Max-SAT solvers with clause learning. In *SAT-2007, Lisbon, Portugal*, pages 28–40. Springer LNCS 4501, 2007.

[8] A. Biere. Picosat essentials. *Journal on Satisfiability, Boolean Modeling and Computation*, 4:75–97, 2008.

[9] Y. Chen, S. Safarpour, A. G. Veneris, and J. P. M. Silva. Spatial and temporal design debug using partial MaxSAT. In *ACM Great Lakes Symposium on VLSI*, pages 345–350, 2009.

[10] S. Darras, G. Dequen, L. Devendeville, and C. M. Li. On inconsistent clause-subsets for max-sat solving. In *CP-2007, Providence/RI, USA*, pages 225–240. Springer LNCS 4741, 2007.

[11] Z. Fu and S. Malik. On solving the partial MAX-SAT problem. In *SAT-2006, Seattle, USA*, pages 252–265. Springer LNCS 4121, 2006.

[12] F. Heras, J. Larrosa, S. de Givry, and T. Schiex. 2006 and 2007 Max-SAT evaluations: Contributed instances. *Journal on Satisfiability, Boolean Modeling and Computation*, 4(2-4):239–250, 2008.

[13] F. Heras, J. Larrosa, and A. Oliveras. MiniMaxSAT: An efficient weighted Max-SAT solver. *Journal of Artificial Intelligence Research*, 31:1–32, 2008.

[14] A. Kuegel. Improved exact solver for the Weighted MAX-SAT problem. In *Proceedings of Workshop Pragmatics of SAT*, 2010.

[15] D. Le Berre and A. Parrain. The Sat4j library, release 2.2. *Journal on Satisfiability, Boolean Modeling and Computation*, 7:59–64, 2010.

[16] C. M. Li, F. Manyà, N. O. Mohamedou, and J. Planes. Resolution-based lower bounds in MaxSAT. *Constraints*, 15(4):456–484, 2010.

[17] C. M. Li, F. Manyà, and J. Planes. New inference rules for Max-SAT. *Journal of Artificial Intelligence Research*, 30:321–359, 2007.

[18] C. M. Li, F. Manyà, Z. Quan, and Z. Zhu. Exact MinSAT solving. In *SAT-2010, Edinburgh, UK*, pages 363–368. Springer LNCS 6175, 2010.

[19] H. Lin, K. Su, and C. M. Li. Within-problem learning for efficient lower bound computation in Max-SAT solving. In *AAAI-2008, Chicago/IL, USA*, pages 351–356, 2008.

[20] V. M. Manquinho, R. Martins, and I. Lynce. Improving unsatisfiability-based algorithms for Boolean optimization. In *SAT-2010, Edinburgh, UK*, pages 181–193. Springer LNCS 6175, 2010.

[21] K. Xu, F. Boussemart, F. Hemery, and C. Lecoutre. A simple model to generate hard satisfiable instances. In *IJCAI-2005*, pages 337–342, Edinburgh, Scotland, 2005.

Lazy Learning Methods for Quality of Life Assessment in People with Intellectual Disabilities

Eva ARMENGOL [a,1], Pilar DELLUNDE [b] and Carlo RATTO [c,2]
[a] *Institut d'Investigació en Intel·ligència Artificial (IIIA-CSIC)*
[b] *Universitat Autònoma de Barcelona*
[c] *Taller Jeroni de Moragas*

Abstract. In this paper we present a preliminary work applying machine learning techniques to the assessment of quality of life (QOL). In 2008 the Government of Catalonia introduced the GENCAT scale, a QOL questionnaire for dependent people users of the social and human services of Catalonia. Using data from a QOL research of 50 people with intellectual disabilities and/or mental illness of the Taller Jeroni de Moragas, we have applied a lazy learning method to discover relations between the different dimensions considered in the GENCAT scale. Our goal is to provide a basis to refine the model of QOL in a way that could support general intervention programs and a better understanding of the necessities of dependent people. This study is an interdisciplinary research of computer scientists, psychologists and human service practitioners.

Keywords. Machine Learning, Lazy learning methods, Psycological applications, Quality of life assessment, GENCAT scale

Introduction

The concept of quality of life (QOL) was introduced into the fields of education, health care, and social services in the early l980s. The notion of QOL includes both objective and subjective factors, and also emic and ethical factors, shared and inherent in all people. This concept allows to focus on a Competence Model that includes principles of self determination (SD) and social inclusion (SI). During the last three decades, the concept of QOL has become fundamental for implementing Person-Centred Planning (PCP), to assess and report personal outcomes, to guide quality improvement strategies, and to evaluate the effectiveness of these practices and strategies.

As a result of the development of this concept of QOL in the long term, in December 2006 the United Nations developed the *Convention on the Rights of Persons with Disabilities* [11]. The Convention is a human rights instrument with explicit SD and SI as

[1]Corresponding Author IIIA-CSIC and Universitat Autònoma de Barcelona: Campus de la UAB, 08193-Bellaterra, Catalonia, Spain; E-mail: eva@iiia.csic.es
[2]Corresponding Author Taller Jeroni de Moragas: C/ Villadelprat 79, 08197-Sant Cugat del Vallès (Barcelona), Catalonia, Spain; E-mail: psicoleg@jmoragas.org

QOL dimensions. Since in the last few years, the implementation of universal treaties had not caused significant changes in the relationship between disability and discrimination [17], this convention, unlike any other, was created with the participation of people with disabilities, and was developed specifically for them as a measure of positive discrimination. The Convention clarifies and qualifies how all categories of rights are applied, identifies areas where adaptations have to be made for the effective exercise of the rights and areas where the rights have been violated, and thus where protection of rights must be reinforced.

The central principles contained in this convention are the same than those that have enriched the change in the last three decades of research in the field of disability since new paradigms were introduced in the study of QOL: disability is placed as a feature within the diversity of human beings and not as a feature to define a person's life. Following this principles, Schalock and Verdugo [1] introduced a multidimensional model of QOL in which the assessment and intervention takes into account what is the situation of the person with disabilities regarding autonomy, SD, SI and also what are the facilities and limitations encountered in the different systems (family, work, home, community, etc.) in order to intervene giving support with the required intensity under each case. Dependent people have often difficulties concerning participation in society, this means that such individuals and their families are already segregated or in danger of being excluded from many situations and opportunities, normally available to other people. The concept of QOL is central in social and labor policy and also in developing and modifying environments that allow access for all individuals to places and resources.

In this paper we have worked with data from the Taller Jeroni de Moragas (TJM), a cooperative of social initiative established in 1973, for people with intellectual disabilities. The TJM has as its primary mission the SI and the autonomy of disabled adults, to achieve that, they elaborate realistic plans of life for people with intellectual disabilities, taking into account their relationship with the world around. The cooperative covers various services: a Supported Employment Centre (SEC), an Occupational Therapy Service (OTS), a Housing Service, a Support Service to Own Home (SSOH) and a Legal Representative Foundation.

In the last years, the TJM has carried out a research in order to obtain an specific (individual) and generic (service) profile of QOL. Integrating the ecological and multidimensional characteristics of the QOL concept, the TJM has applied the GENCAT scale (see next section for the details on this scale) to assess the QOL. The results have served to implement the PCP's interventions required according to the specific needs of each case. The outcomes of this research have been useful also to check the effectiveness of a new evaluation procedure. The overall results were used to propose practices to improve the dimensions more influent in the global QOL of every individual.

The study of QOL assessment from the point of view of Artificial Intelligence differs from other known domains. Specifically, the QOL model of Schalock and Verdugo [1] considers both subjective and objective variables affecting an individual's QOL. Far from considering this a serious drawback, we regard it as an opportunity, both from a theoretical and from a practical point of view. Our work plans to be a contribution to a more adequate knowledge representation of domains in social sciences. Taking as our starting point the TJM's research, what we propose here is to perform knowledge discovery on the QOL domain using lazy learning methods (as we proposed in [16] for

Table 1. Dimensions and core indicators of quality of life.

DIMENSIONS	INDICATORS
Emotional well-being (EW)	Satisfaction. Self-concept. Lack of stress/negative feelings.
Interpersonal relations (IR)	Social, familiar and affective relationships. Have stable and clearly identified friends. Positive and Gratifying social contacts. Satisfying sex life.
Material well-being (MW)	Housing conditions. Workplace conditions. Service conditions. Employment Incomes/salary. Possessions.
Personal development (PD)	Education. Learning opportunities. Work and Functional abilities. Activities of daily living. Access to new technologies.
Physical well-being (PW)	Health care access. Health care consequences (sorrow, medication, etc.) Functional diet, sleep, mobility. Technical assistance.
Self-determination (SD)	Autonomy. Goals and personal preferences. Decisions. Choices.
Social inclusion (SI)	Participation. Real integration. Access. Supports.
Rights (RI)	Knowledge of rights. Defense of rights. Exercise of rights. Privacy. Respect.

other domains). The idea is to use a lazy method for classifying a given problem and to generate a domain model from the explanation of the classification.

The structure of this paper is the following. First we briefly explain the multidimensional QOL Model and the GENCAT scale. In section 2 we introduce the LID method and how a domain theory is generated by it. In Section 3 we describe the experiments and discuss the results. Finally we present our conclusions and future work.

1. The QOL Model and the GENCAT scale

In the last thirty years, the definitions of the American Association on Intellectual and Developmental Disabilities (AAIDD) (see [5], [6] and [10]) have evolved from a concept of disability as an static trait inherent to the individual (based on the strict measurement of the intellectual quotient) to a new analysis (see [7], [8] and [9]) that bases disability on the interaction between the skills (performance competence) of people with disabilities and the support (integration facilities) of their context. The influence of this transfer (from a strictly psychometric concept of disability to a new kind of practical and social intelligence) and the development of several cross-cultural research, has determined the currently QOL model, composed of eight dimensions and core indicators (listed in Table 1) that define operationally each QOL dimension.

Besides this multidimensional perspective, Schalock and Verdugo [1] also use an ecological (i.e. systems) perspective that includes:

1. microsystem: the immediate social settings, such as family, home, peer group,
2. mesosystem: the neighborhood, community, service agencies, and organizations that directly affect the person's life
3. macrosystem: the overarching pattern of culture, social-political trends, economic systems, and society-related factors that directly affect one's values, assumptions, and the meaning of words and concepts.

In 2008 the Catalonian Institute of Assistance and Social Services (ICASS) of the Government of Catalonia, and the Institute on Community Integration (INICO) of the

```
Function LID (p, S_{D_i}, D_i, C)
    if stopping-condition(S_{D_i})
        then return class(S_{D_i})
        else f_d := Select-attribute (p, S_{D_i}, C)
            D_{i+1} := Add-attribute(f_d, D_i)
            S_{D_{i+1}} := Discriminatory-set (D_{i+1}, S_{D_i})
            LID (p, S_{D_{i+1}}, D_{i+1}, C)
    end-if
end-function
```

Figure 1. The LID algorithm: p is the problem to be solved, D_i is the similitude term, S_{D_i} is the discriminatory set associated with D_i, C is the set of solution classes, $class(S_{D_i})$ is the class $C_i \in C$ to which all elements in S_{D_i} belong.

University of Salamanca, introduced the GENCAT scale (see [2] and [3]). This scale is a QOL questionnaire for dependent people users of the social and human services of Catalonia. It includes people with intellectual disabilities, elder people and also people with physical disabilities, mental disease, drug dependence or AIDS/HIV. The construction of this assessment tool is supported by the theoretical framework of the latest definitions of the AAIDD and the QOL model proposed by Schalock and Verdugo in [1]. The psychometric properties of the GENCAT scale were analyzed after applying the instrument to a representative sample of 608 professionals and 3,029 users of 239 services from Catalonia (the instrumental study that validates this questionnaire is presented in [4]).

2. Generating a domain model with the Lazy Induction of Descriptions Method

Lazy Induction of Descriptions (LID) is a lazy learning method for classification tasks. LID determines which are the most relevant attributes of a new problem and searches in a case base for cases sharing these relevant attributes. The problem is classified when LID finds a set of relevant attributes whose values are shared by a subset of cases all of them belonging to a same class. The description formed by these relevant features is called *similitude term* and the set of cases satisfying the similitude term is called *discriminatory set*.

Given a problem for solving p, the LID algorithm (Fig. 1) initializes D_0 as a description with no attributes, the discriminatory set S_{D_0} as the set of cases satisfying D_0, i.e., all the available cases, and C as the set of solution classes into which the known cases are classified. Let D_i be the current similitude term and S_{D_i} be the set of all the cases satisfying D_i. When the stopping condition of LID is not satisfied, the next step is to select an attribute for specializing D_i.

The specialization of D_i is achieved by adding attributes to it. Given a set F of attributes candidate to specialize D_i, LID selects the most discriminatory attribute in F using a distance measure. Such distance is used to compare each partition $\mathcal{P}_f$ induced on S_{D_i} by an attribute f with the correct partition $\mathcal{P}_c$. The *correct partition* has as many sets as solution classes. Each attribute $f \in F$ induces in S_{D_i} a partition $\mathcal{P}_f$ with as many sets as the number of different values that f takes in the cases contained in S_{D_i}. Given a distance measure Δ and two attributes f and g inducing respectively partitions

$\mathcal{P}_f$ and $\mathcal{P}_g$, we say that f is *more discriminatory* than g iff $\Delta(\mathcal{P}_f, \mathcal{P}_c) < \Delta(\mathcal{P}_g, \mathcal{P}_c)$. This means that the partition $\mathcal{P}_f$ is closer to the correct partition than the partition $\mathcal{P}_g$. In our experiments, the selection of the most discriminatory attribute is heuristically done using the LM distance [12] over the candidate attributes.

Let f_d be the most discriminatory attribute in F. The specialization of D_i defines a new similitude term D_{i+1} by adding to D_i the attribute f_d. The new similitude term $D_{i+1} = D_i \cup \{f_d\}$ is satisfied by a subset of cases in S_{D_i}, namely $S_{D_{i+1}}$. Next, LID is recursively called with $S_{D_{i+1}}$ and D_{i+1}. The recursive call of LID has $S_{D_{i+1}}$ instead of S_{D_i} because the cases that are not satisfied by D_{i+1} will not satisfy any further specialization. Notice that the specialization reduces the discriminatory set at each step, i.e., we get a sequence $S_{D_n} \subseteq S_{D_{n-1}} \subseteq \ldots \subseteq S_{D_0}$.

LID has two stopping situations: 1) all the cases in the discriminatory set S_{D_j} belong to the same solution class C_i, or 2) there is no attribute allowing the specialization of the similitude term. When the stopping condition 1) is satisfied p is classified as belonging to C_i. When the stopping condition 2) is satisfied, S_{D_j} contains cases from several classes; in such situation the *majority criteria* is applied, and p is classified in the class of the majority of cases in S_{D_j}. The outcome of LID is a class and the last similitude term, that justifies the proposed classification of the new problem.

The similitude term can be interpreted in several ways. A possible interpretation is that the similitude term can be seen as a partial discriminant description of C_i since all the cases satisfying the similitude term belong to C_i (according to one of the stopping conditions of LID). Therefore, the similitude term can be used as a generalization of knowledge in the sense of either PROTOS [13], EBL or inductive learning methods. Since the similitude term contains the important attributes used to classify a problem, it can also be interpreted as a justification or *explanation* of the problem classification. Notice that in any of the interpretations above, LID explanations can be taken by the system as domain rules since they contain the relevant attributes for classifying a problem. What we propose is to use LID with leave-one-out and to store the explanations of the classifications to form a domain theory. In previous works [14], [15] and [16], we stored only the explanations corresponding to correct classifications of LID. However, in this domain we store all the explanations due to the subjectivity part of the classifications given by the GENCAT scale. Our point is that an explanation of an incorrect classification of LID could show where are the subjectivities of the scale and how to correct them.

3. Experiments

We have experimented with a case base composed of descriptions of 50 persons from the TJM, 33 participants with intellectual disability and/or mental illness of the SEC (out of 64 members of this center) and 17 from the OTS (out of 28 persons in this service), all of them adults and workers of TJM with an average age of 43 years old. Each person is described using the 8 dimensions of the model proposed by Schalok and Verdugo (see Table 1). Each one of the dimensions has a numerical value obtained from the answers of the questionnaire mentioned in [4] (this questionnaire is the basis of the GENCAT scale). In our experiments we have discretized the values of each dimension according to the intervals provided by the experts (psychologists and human service practitioners). The aggregation of the 8 dimensions gives the global index of quality of life (IQV) that has also been discretized according the expert's assessment (see Table 2).

Table 2. Discretization of the values of both the dimensions and the IQV.

	Interval	Label		Interval	Label
dimensions	1- 6	very-low	IQV	52 - 85	very-low
	7 - 9	low		86 - 95	low
	10 - 11	medium		96 - 105	medium
	12 - 13	high		106 - 115	high
	14 - 20	very-high		116 - 138	very-high

Experts are interested in knowing which kind of strategies could be done in order to improve the IQV, to achieve this goal it is important to know the interrelations among the different dimensions. This kind of study has not received many attention until now, although there are some hypotheses about these interrelations. Specifically, in the particular case of the TJM, the experts want to analyze in depth the relationship of RI, SI and SD with the other dimensions and also with the IQV. This particular goal was established after a previous statistical analysis made by the TJM's professionals. The results obtained showed that the mentioned dimensions are those having lower indexes and, therefore the ones where it is necessary to put emphasis in order to improve the QOL of the workers of the TJM.

We have used the LID method with leave-one-out to obtain descriptions of the classes, for instance what defines a low (medium, high or very high respectively) IQV. The conducted experiments focus on generating a domain theory taking as solution classes the values of IQV, RI, SI and SD. Each one of these experiments gives us interesting information that has been contrasted with the expert's knowledge on the psychological and social areas of people with disabilities. In the following we present the domain theory obtained by LID and a discussion of the results taking into account the input data. We want to remark that this preliminary study cannot be taken as a general study because it has not statistical validity due to the size of the database. However, it has provided to the experts a useful profile of the TJM and, in addition, it has served as basis to further discussions of particular cases that do not completely agree with the rules obtained. We expect also that the discordance issues between the LID theory and the expert's model would be a start point to improve the current model of QOL.

3.1. LID theory to assess the global index of quality of life (IQV)

Figure 2 shows the domain theory generated by LID to assess the IQV. The expert analyzed one by one these rules. Rule 1 completely agrees with the expert's knowledge since a high IQV is correlated with a functional social life. In other words, having high IR means having social interaction functional abilities and thus, the person has the possibility to increment the SD. Also, a high RI is a signal of inclusion in the society. Rule 2 needs some discussion because it implies some aspects that are not clear from the expert's point of view. For instance, experts point out that a high IQV is a consequence of a SD high or very high, although lower levels of SD should be compensated by high values of SI, PW or RI. Therefore, this rule should be analyzed and discussed in more depth.

Rule 3 agrees with the expert's knowledge because a healthy person, having a high PW, adaptive abilities and social interaction abilities, has also a high probability of having a very high life quality. Rule 4 also fits the expert's knowledge since it confirms the

RULES TO ASSES THE IQV

1) (HIGH) when "(RI HIGH)" and "(IR HIGH)"
2) (HIGH) when "(SD MEDIUM)" and "(IR VERY-HIGH)"

3) (VERY-HIGH) when "(PW HIGH)" and "(PD VERY-HIGH)" and "(IR VERY-HIGH)"
4) (VERY-HIGH) when "(SD HIGH)" and "(IR VERY-HIGH)"
5) (VERY-HIGH) when "(PW VERY-HIGH)" and "(IR VERY-HIGH)"

6) (MEDIUM) when "(MW HIGH)" and "(IR MEDIUM)"

Figure 2. Rules generated by LID to assess the IQV.

knowledge already mentioned in Rule 2: a person with high SD commonly has also high social abilities (IR), and both emotional and physical well-being (EW and PW) allowing him to take his own decisions. Concerning Rule 5, although it does not fit exactly the expert's knowledge, the inverse relation seems to be true, that is to say: a low PW commonly is associated with low SI and RI and, thus, with a low IQV. Both dimensions, PW and IR joined with EW are factors that influence the other dimensions and, consequently, the IQV.

3.2. Correlation between SD and RI

The results taking RI as solution class show that this dimension is correlated with SD, since this last dimension is considered in all the rules produced by LID (see Figure 3). As we can see in Figure 4 (the input cases with a *high* assessment of SD), the values of SD and RI coincide except in three cases: 22, 24 and 41. Notice that, in these three cases, the assessment of PD is *high* or *medium*. This could mean that the PD could also be a factor to be taken into account in high assessments of SD.

RULES TO ASSESS RI

(HIGH) when (PD MEDIUM) and (SD MEDIUM)
(HIGH) when (SI VERY-HIGH) and (SD HIGH)
(HIGH) when (SD VERY-HIGH)
(HIGH) when (RI VERY-HIGH) and (SD HIGH)
(HIGH) when (RI VERY-HIGH) and (SD MEDIUM)
(HIGH) when (SI HIGH) and (SD MEDIUM)
(VERY-HIGH) when (SD VERY-HIGH)

(LOW) when (PD HIGH) and (SD LOW)
(LOW) when (PD LOW) and (SD LOW)
(LOW) when (PD MEDIUM) and (SD MEDIUM)
(LOW) when (PD VERY-HIGH) and (SD LOW)

(MEDIUM) when (PW HIGH) and (PD MEDIUM) and (SD LOW)
(MEDIUM) when (SD VERY-LOW)
(MEDIUM) when (SI VERY-HIGH) and (SD MEDIUM)
(MEDIUM) when (PD HIGH) and (SD HIGH)
(MEDIUM) when (PD HIGH) and (SD LOW)
(MEDIUM) when (PD VERY-HIGH) and (SI VERY-HIGH) and (SD MEDIUM)

Figure 3. LID rules for assessing RI.

A direct correlation between SD and RI is not so clear for lower assessments of SD. Thus, when SD is *low* an important dimension appearing in most of the LID rules is the

PW whereas for the *medium* assessment of SD there are a lot of dimensions taken into account in the rules generated by LID. Clearly, the size of the database does not allow to make more accurate hypotheses on these relations.

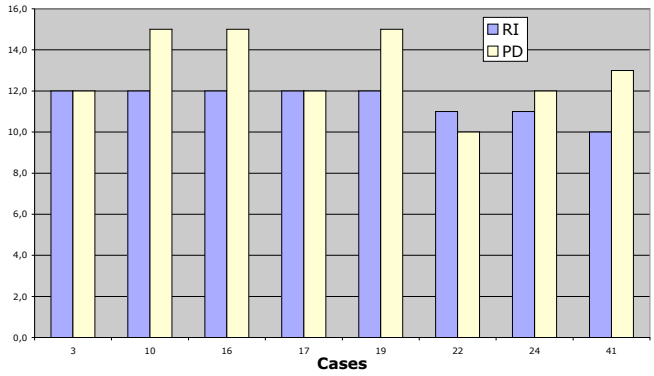


Figure 4. Values of RI and PD for the input cases having SD = *high*.

3.3. Rules generated by LID to assess high SI

Figure 5 shows the rules generated by LID to assess SI as high. The expert does not completely agree with Rules 1 and 2. His main concern is that low levels of RI and EW, respectively, can not produce high SI. In particular, a low EW does not facilitate social relations and if, as it is stated in Rule 2, IR is also low (i.e. the person has not many relational abilities), then the RI has a low probability to be high.

1) (HIGH) when "(RI LOW)" and "(IR MEDIUM)"
2) (HIGH) when "(EW LOW)" and "(IR LOW)"
3) (HIGH) when "(PW VERY-HIGH)" and "(IR HIGH)"
4) (HIGH) when "(RI HIGH)" and "(IR HIGH)"
5) (HIGH) when "(RI HIGH)" and "(PW VERY-HIGH)" and "(IR HIGH)"

Figure 5. LID rules for *high* SI.

Rule 3 does not completely satisfy the expert, however, it seems to be reasonable since it is very possible that a very high PW and a high IR could motivate high IS. Rule 4 is very similar to Rule 3 but it involves RI instead of PW. Rule 5 is, in the expert's opinion, the best rule among the rules obtained by LID to define high IS, because it involves both PW very high and RI high, meaning that combining a good physical state with a good knowledge of rights can facilitate an increase in the value of the IS.

However, it is worth to point out that none of these rules satisfies completely the expert. A high SI is expected to be accompanied with high PW, IR and SD. This is a

good aspect to be discussed in the future in order to improve, if necessary, the current model of QOL.

4. Concluding Remarks

In this paper we have generated a domain theory using the LID method with data coming from the research of the TJM in which they assessed the QOL through the GENCAT scale. Then, we have used the description of the theory obtained by LID to support the assessment process made by social practitioners in order to improve the QOL of persons with intellectual disabilities. Due to the size of the database we have used (50 descriptions of persons) this is only a preliminary work oriented to prove the feasibility of the method we propose.

Although the sample was not enough representative of the collective of persons with intellectual disabilities, it has helped us to give a good profile of the workers of the TJM. Our priority now is to obtain a bigger database representative of all this group. For this purpose we need to involve the main associations of the sector in order to collect the data.

The GENCAT scale is not specific for persons with intellectual disabilities, it is commonly applied to other users of the social services (elder people, people with physical disabilities, mental diseases, drug dependences or AIDS/HIV). Thus, a natural step in our future research will be to work on another database obtained of the application of the scale to other collectives. An institutional help is essential to fulfil this goal.

In the present study we have worked only with one questionnaire for each person. The procedure created by the technical team of TJM assess and averages, jointly with the direct care staff, the perceptions of the most significantly life areas of people with disabilities (family, work, home and finally, the area of the own perceptions of the worker with disability) using different questionnaires for each person. This kind of procedure minimizes the subjectivity of the evaluators and also allows the effective empowerment of the person with disability. The results obtained underline which dimensions of QOL are better developed and which are most affected in a common trend from the perspective of each of the mainly areas. Further research will include the use of data coming from different questionnaires. An interesting open problem is how to aggregate and weight all this information.

Finally, we would like to carry on an exhaustive comparison of our work with the results obtained using different techniques, for instance, traditional statistical analysis procedures as regression. In future versions of this work we will include mesures to evaluate the LID rules.

Acknowledgements

The authors thank all the professionals of the Taller Jeroni of Moragas and F. Almacelles, X. Canals, A. García-Cerdaña, R. Herbera, J.P. Gonzalez, A. Ruiz, F. Sant, C. Saez, Ll. Vallès and M. Vásquez for their contributions to the project presented in this paper. We would like also to thank the Federation DINCAT and the ICASS for their collaboration. Research partially funded by the Spanish MICINN projects

ARINF (TIN2009-14704-C03-03) and CONSOLIDER (CSD2007-0022), by the ESF Eurocores-LogICCC/MICINN (FFI2008-03126-E/FILO), and the grants 2009-SGR-1433/1434 from the Generalitat de Catalunya.

References

[1] R.L. Schalock and M.A. Verdugo, Handbook of quality of life for human service practitioners, *American Association on Mental Retardation*, Washington, DC, 2002.

[2] M.A. Verdugo, B. Arias, L.E. Gómez and R.L. Schalock, Formulari de l'Escala GENCAT de Qualitat de Vida. Manual d'aplicació de l'Escala GENCAT de Qualitat de Vida, *Departament d'Acció Social i Ciutadania, Generalitat de Catalunya*, Barcelona, 2008, http://inico.usal.es/ documentos/EscalaGencatManualCAT.pdf

[3] M.A. Verdugo, B. Arias, L.E. Gómez and R.L. Schalock, Informe sobre la creació d'una escala multidimensional per avaluar la qualitat de vida de les persones usuàries dels serveis socials a Catalunya, *Departament d'Acció Social i Ciutadania, Generalitat de Catalunya*, Barcelona, 2008, http://inico.usal.es/documentos/EscalaGencatFormularioCAT.pdf

[4] M.A. Verdugo, B. Arias, L.E. Gómez and R.L. Schalock, Development of an objective instrument to assess quality of life in social services: Reliability and validity in Spain, *International Journal of Clinical and Health Psychology*, **10**, 105–123, 2010.

[5] R. Luckasson, S. Borthwick-Duffy, W. H. E., Buntix, D. L. Coulter, E.M. Craig and A. Reeve, Mental Retardation. Definition, Classification, and Systems of Supports, *American Association on Mental Retardation*, 10th ed., Washington, DC, 2002.

[6] R. L. Schalock, S. Borthwick-Duffy, V. Bradley, V., W. H. E., Buntix, D. L. Coulter and E.M. Craig, Intellectual disability. Definition, Classification, and Systems of Supports, *American Association on Intellectual and Developmental Disabilities*, 11th ed., Washington, DC, 2010, www.aaidd.org/index.cfm

[7] M.A. Verdugo, El cambio de paradigma en la concepción del retraso mental: la nueva definición de la AAMR, *Siglo Cero, Revista Española sobre Discapacidad Intelectual*, **25** (5), 5–24, 1994.

[8] M.A. Verdugo, Aportaciones de la definición de retraso mental (AAMR, 2002) a la corriente inclusiva de las personas con discapacidad, 2003, sid.usal.es/idocs/F8/FDO6569/verdugo.pdf

[9] M.A. Verdugo and R.L Schalock, Últimos avances en el enfoque y concepción de las personas con discapacidad intelectual, *Siglo Cero, Revista Española sobre Discapacidad Intelectual*, **41** (4), 7– 21, 2010.

[10] R. Luckasson, D. A. Coulter, E. A. Polloway, Mental retardation: Definition, classification, and systems of supports, *American Association on Mental Retardation*, 9th ed., Washington, DC, 1992.

[11] Convention on the Rights of Persons with Disabilities, *United Nations*, 2006, www.un.org /disabilities/default.asp?id=150

[12] R. López de Mántaras, A distance-based attribute selection measure for decision tree induction, *Machine Learning* **6**, 81–92, 1991.

[13] E.R. Bareiss, B.W. Porter and C.C. Wier, PROTOS: an examplar-based learning apprentice, *Int. J. Man-Mach. Stud.*, **29**(5), 549–561, 1988.

[14] E. Armengol and E. Plaza, Lazy Learning for Predictive Toxicology based on a Chemical Ontology, *Artificial Intelligence Methods and Tools for Systems Biology*, Kluwer Academic Press. W. Dubitzky and F.J. Azuaje (eds) 2004.

[15] G. Corral, E. Armengol, A. Fornells and E. Golobardes, Explanations of Unsupervised Clustering applied to Data Security Analysis, *Neurocomputing*, **72**, 2754-2762. 2009

[16] E. Armengol, Building partial domain theories from explanations. *Knowledge Intelligence*, **2**(8), 19–24, 2008.

[17] G. Quinn and T. Degener, Derechos Humanos y Discapacidad, *Oficina del Alto Comisionado de Naciones Unidas, Naciones Unidas*, Nueva York y Ginebra, 2002.

[18] C. Ratto, Avaluació i intervenció en els diferents àmbits de la qualitat de vida de les persones amb discapacitat intel·lectual i/o malalties mentals. I Jornada sobre Intel·ligència Artificial i Qualitat de Vida, 2010, http://jornades.uab.cat/qualitat/

Probabilistic Appearance-based Mapping and Localization using the Feature Stability Histogram

Bladimir BACCA[a,b,1], Joaquím SALVÍ[a], Xavier CUFÍ[a]
[a] *Universitat de Girona, Girona, Catalunya, Spain*
[b] *Universidad del Valle, Cali, Colombia*

Abstract. Appearance-based methods for mapping and localization have gained increasing attention in recent years. The strength of these models lies in their ability to represent the environment through high-level image features. However, the environment illumination, occlusions and walking people have a negative impact on these approaches. This paper presents a probabilistic appearance-based mapping and localization approach which uses the Feature Stability Histogram to update the environment appearance continuously and extract the more stable features in the environment. Our proposed method uses these stable features as successive appearance measurements to update the posterior probabilities incrementally on a topological map using a Rao-Blackwellized particle filter. Our algorithm considers omnidirectional images and laser data as measure of the environment appearance. Our approach was evaluated on a robot in a dataset collected along various seasons and time of day.

Keywords. Appearance-based, topological mapping, localization, particle filters.

Introduction

Reliable self-localization and map building plays an important role for autonomous navigation in mobile robotics. In recent years, appearance-based mapping and localization has gained special attention since these methods use a richer description at certain points in the environment, which allow them to get a better global impression of the surroundings and this description gives more cues to improve robot mapping and localization. The latter gives to appearance-based methods promising new directions in comparison with the strong geometric feature based methods. Many solutions have been presented in recent years for the appearance-based mapping and localization problem. The review presented in this work is focused on those works performing concurrent mapping and localization, or those considering a probabilistic mapping and localization framework, and those works with alternative environmental representations.

Earlier works for appearance-based mapping and localization were focused on explicitly comparing image templates, which gave coarse position estimates. A first attempt to include probabilistic pose estimates using appearance-based models was presented in [1] and [2]. The former proposed a concurrent mapping and localization approach using a Multi-Gaussian position representation and reducing the uncertainty in position through the covariance intersection rule when a place is re-visited. In [2] the Dempster-Shafer probability theory was applied to topological maps in order to

1 Corresponding Author: University of Girona, Campus Montilivi, Building P-IV, Girona, Spain, ZIP 17071. E-mail: bladimir@eia.udg.edu, evbacca@univalle.edu.co.

estimate the robot position, where the appearance of the environment was modeled using invariant column segments with color enhanced SIFT descriptors. Another work which combines appearance-based methods with well known Simultaneous Localization and Mapping (SLAM) approaches is presented in [3], where vertical edge features were considered to perform bearing SLAM.

An alternative approach was proposed adapting an algorithm developed for fast and accurate image search. Images can be compared very quickly to see if they have many features in common, and hence may be of the same object or location. This algorithm is known as Bag-of-words (BOW) and it is used for appearance-based mapping and localization in [4-6]. The BOW algorithm was used in [4] as loop-closure detection, a particularly hard problem and important component of the SLAM problem. In [4] the authors suggest a method based on the BOW approach to compare the local appearance of the environment at regular intervals, then sequentially build a similarity matrix which is used to extract loop-closure hypotheses. The visual words dictionary used in [4] has to be created in an off-line process. However, in [5] this part of the process is done incrementally, and an efficient optimization algorithm is used to integrate odometry and generate consistent topo-metrical map. Recently, the BOW method was completely integrated in a SLAM algorithm [6].

With respect to probabilistic appearance-based methods, the FAB-MAP 2.0 approach [7] proposes a probabilistic SLAM approach based on BOW. It captures the feature dependencies within one scene using Chow-Liu trees. The FAB-MAP algorithm has proven successful in large-scale environments (outdoor and indoor). State estimators based on Rao-Blackwellized particle filters (RBPF) were used in [8] and [9]. The former estimates 2D position and orientation as well as the graph representation of the environment, which is indoor and with high perceptual aliasing. The latter uses a RBPF approach to estimate the posterior probabilities on the space of all possible topologies given a set of appearance measurements. Other ways to solve the appearance-based mapping problem are presented in [10] and [11]. In [10] an interesting method is proposed based on planar regions to extract the environment topology using a single camera. And, in [11] a novel concept called augmented spherical view is proposed to estimate 6DOF poses using a hybrid laser range/omnidirectional image sensor.

This paper extends the Feature Stability Histogram (FSH) presented in [12], where an innovative feature management method was proposed. This method is based on the Atkinson and Shiffrin memory model [13]. The feature management is focused on distinguish stable features, LTM (Long-Term-Memory) features, and unstable features, STM (Short-Term-Memory) features. This paper describes a probabilistic framework which uses the FSH to obtain an appearance-based concurrent mapping and localization algorithm. Our approach uses a topological map represented in a graph and a RBPF to estimate the node position given laser scan and omnidirectional images as measurements.

This work is organized as follows: Section 1 describes how the FSH is related with our topological map structure. Section 2 describes our proposed probabilistic model. Section 3 presents the experimental results obtained and Section 4 our conclusions.

1. Topological mapping and the Feature Stability Histogram

For years scientific community has been finding inspiration in nature. Probabilistic robotic models have their origins in how the "place cells" in the hippocampus work. In our case, the Atkinson and Shiffrin memory model [13] is used to distinguish stable features from unstable ones, then using the stable features for robot mapping and localization. The Atkinson and Shiffrin memory model has four main components: the Short-Term Memory (STM) which retains information long enough to use it, the Long-Term Memory which retains information for longer periods of time or lifetime, the Sensory Model which is the capability of the sensing organs to discriminate information for subsequent processing, and the forgetting module which affects all other components since it was experimentally demonstrated that memories can be forgotten through a trace decay. This model proposes stimuli inputs enter in the STM, and then if these inputs are continuously rehearsed, they become part of the LTM. Information retained in LTM is recalled continuously in lifetime, but it does not reside permanently, if this information is not rehearsed it can be forgotten. The memory model proposed in [13] has drawn criticism from psychologists and neuroscientists due its extreme linearity of the memory process [14-15]. They argue that the Atkinson and Shiffrin model does not take into account the ability of many people to recall information despite the fact this information has not been rehearsed. In other words, apparently stimuli inputs can reach LTM bypassing STM. In addition, this memory model does not consider different levels of memory [14-15], which in our case it would be useful to take into account the strength of the image features.

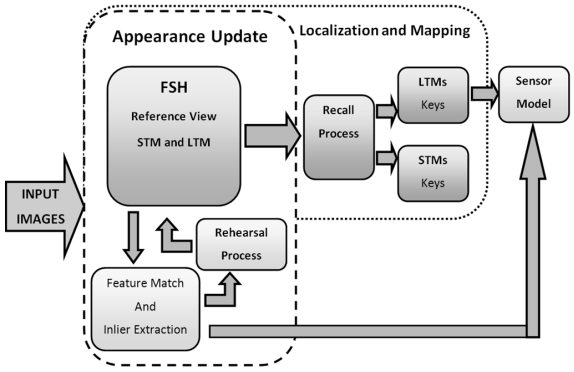


Figure 1. Our appearance-based update model.

In this work, we use our appearance-based update approach for robot mapping and localization inspired in the Atkinson and Shiffrin memory model and depicted in Figure 1. In first place, the reference view is composed by both memories: the STM and LTM. It has two main advantages: first, an input feature can bypass the STM and become a LTM keeping in mind the feature strength e.g. the feature uncertainty, the Hessian value in the SURF descriptor, or the matching distance; second, using the FSH as reference view the feature classification as STM or LTM is not linear since the rehearsal process can take into account the feature strength. The rehearsal process is based on the number of times a feature has been observed, but weighted by a function which depends on the matching distance computed through robust RANSAC outlier rejection and epipolar geometry constraint. In this way, the appearance of the

environment represented for the FSH is updated according to the presence or absence of pre-observed features, or the inclusion of new features whose vote values are weighted by the normalized Hessian value in our case. The recall process is in charge of distinguishing between STM and LTM features, i.e. differentiating the most stable features (LTM) from the STM features. A feature descriptor can be defined as an LTM feature if it has a high value in the FSH; otherwise it is considered as a STM feature. This classification has two main advantages: first, it is a straightforward method to deal with temporal occlusions because using the weighted voting scheme of the rehearsal process, the FSH value of the corresponding feature suffers a relative decrease, or it will increase if the feature is re-observed; second, it is a suitable method to deal with changing environments where illumination changes and pedestrians cause feature appearance or disappearance, and in the end the more stable features will belong to the LTM and those features only are used for mapping and localization. The recall process implemented in our approach is threshold-based, i.e. FSH values greater than a threshold are considered LTM features, and those less than a threshold are STM features [12]. Once the LTM features are found, they are used to build the sensor model in a Bayesian framework for mapping and localization. The sensor model proposed uses a similarity measure based on the ratio of inliers found and the total number of LTM features of the reference view. The joint work done by the rehearsal and recall processes allows the appearance update of the environment and to obtain a sensor model for mapping and localization.

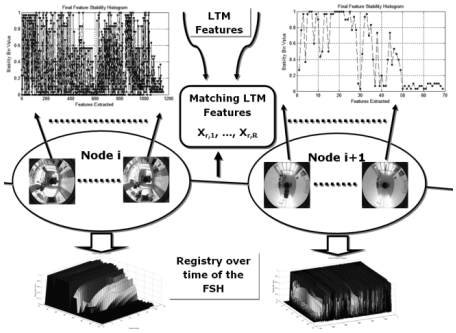


Figure 2. Internal structure of the topological map considering the feature stability histogram.

Topological maps are compact, consume less computer memory, can be stored in efficient data structures, and are able to hold high-level information that can be used for semantic environmental modeling. Our topological map (a graph based representation) is composed of several nodes, each of which stores one or more omnidirectional views. The structure of our topological map representation is illustrated in Figure 2, and it takes into account the following notation:

- A node is defined as n_i, $i \in \{1, ..., N\}$ where N is the number of nodes in the map. This node represents a place in the environment which 2D position estimate is obtained from scan matching.
- A node also stores a set of SURF descriptors extracted from similar views, which are denoted by $Dn(i,j)$ where i is the node index, and $j \in \{1, ..., K\}$. Here, K is the number of feature descriptors stored within a node.

- A node in the topological map stores its own FSH built from the above descriptors. It is denoted as *fsh(i,t), i ∈ {1, ..., N}* at time *t*, where *t* denotes the number of time stamps the FSH has been updated.
- Edges between nodes define neighboring relationships using scan matching and they store a set of corresponding features extracted from a two-view geometry process between nodes and denoted by Ed_r, $r \in \{1, ..., R\}$ where *R* is the number of edges between nodes.

2. Topological mapping using Rao-Blackwellized Particle Filters

In this work, we are interested in finding the posterior of the probability distribution over the topological nodes $p(n_t \mid s_t, a_t)$, where n_t is the node in the topological map at time *t*, s_t is the laser scan at time *t*, and a_t is the appearance of the environment given by a set of SURF descriptors [16] at time *t*. Then, applying the Bayes law the posterior probabilities can be expressed in terms of the measurement likelihood and prior probabilities as depicted in Eq. (1), where the prior on the node position at time *t* can be written as Eq. (2) shows.

$$p(n_t|s_t, a_t) \propto p(s_t, a_t|n_t, s_{t-1}, a_{t-1})p(n_t|s_{t-1}, a_{t-1}) \quad (1)$$

$$p(n_t|s_{t-1}, a_{t-1}) = \int p(n_t|n_{t-1}, s_{t-1}, a_{t-1})p(n_{t-1}|s_{t-1}, a_{t-1})dn_{t-1} \quad (2)$$

$$p(n_t|s_{t-1}, a_{t-1}) \cong \int p(n_t|n_{t-1})p(n_{t-1}|s_{t-1}, a_{t-1})dn_{t-1} \quad (3)$$

where $p(n_{t-1} \mid s_{t-1}, a_{t-1})$ gives the recursive formulation in order to keep in mind the past estimates, and $p(n_t \mid n_{t-1}, s_{t-1}, a_{t-1})$ is the prior of the node position which in our work is considered as the proposal distribution in the particle filter framework. Assuming that the system dynamics is Markovian, which means that future locations do not depend on past locations, this prior can be approximated to $p(n_t \mid n_{t-1})$ as depicted in Eq. (3). The latter distribution is called the motion model, and it is estimated using scan matching, since it involves metric information. The particle filter maintains the sampled posterior over the node locations, given the appearance of the environment (laser scans and SURF descriptors). The motion model is sampled from the previous belief, and then a new *i-th* state n_t^i is obtained from $p(n_t^i|n_{t-1}^i)$. Each sample, or particle, has an importance weight which is defined as the ratio of the target distribution to the proposal distribution as it is shown in Eq. (4).

$$w_t^i = \frac{p(n_t|s_t,a_t)}{p(n_t|s_{t-1},a_{t-1})} w_{t-1}^i \propto p(s_t, a_t|n_t^i, s_{t-1}, a_{t-1}) w_{t-1}^i \quad (4)$$

where, as target distribution we used the Eq. (1) and as proposal distribution we used Eq. (2), w_t^i is the updated *i-th* weight of the particle *i-th* and w_{t-1}^i is its weight at time *t-1*. We assumed the image appearance measures given by the SURF descriptors and the laser scan measures are conditionally independent each other, then the $p(s_t, a_t \mid n_t, s_{t-1}, a_{t-1})$ can formulated in two independent distributions as Eq. (5) shows.

$$p(s_t, a_t|n_t^i, s_{t-1}, a_{t-1}) = p(s_t|n_t^i, s_{t-1}) p(a_t|n_t^i, a_{t-1}) \quad (5)$$

where, $p(s_t|n_t^i, s_{t-1})$ likelihood is computed using the Hausdorff fraction between laser scans around the last estimated position, and the current laser range measures. The $p(a_t|n_t^i, a_{t-1})$ likelihood is computed using the similarity measure depicted in Eq. (6) between the LTM features of the nodes around the last estimated position, and the current SURF features.

$$H_t^{i,j} = \frac{M_t^{i,j}}{\sqrt{M_t^{i,j} D_{curr}}} \tag{6}$$

where, $M_t^{i,j}$ is the number of matches between the LTM features of node *j* and the current SURF descriptors D_{curr}, in the particle *i* at time *t*. The matching procedure considers a robust RANSAC outlier rejection and two-view geometry constraint.

An underlying advantage of our proposal is that it gives two cues to obtain loop closing hypotheses. The main idea behind this is depicted in Figure 3, where given a topological map with 2D metric information inferred from scan matching, the laser scan measurements and the environmental appearance (SURF descriptors), three filters are in charge of generating a loop closing hypotheses. The first one is a position-based filter; basically it tests if the actual node location is close to a previously mapped node, as Eq. (7) shows. If so, a set of nodes specified by [18] and centered in one with the minimum distance are passed to the next filter. The second filter takes into account the laser scans. The Hausdorff fraction is computed to get the laser scans measurement likelihood, as Eq. (8) shows. If the second filter has meaningful results, the LTM features of the selected nodes are tested against to the current features using robust RANSAC outlier rejection and two-view geometry constraints to obtain the image measurement likelihood. Finally, since the laser scans and image appearance information are conditionally independent, the last two likelihood measurements are combined in the same way as Eq. (5) shows to obtain a loop closing hypotheses and used to reconfigure the topological representation of the environment using [17].

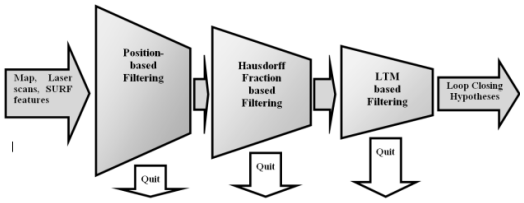


Figure 3. Loop closing hypotheses generation.

$$n_s = min_k\left[\left\|x_t - x_{t-1}^k\right\|\right] \tag{7}$$

$$p_{LC} = max_k\left(p(s_k)p(i_k)\right) = max_k\left(HF\left(s_{n_s}^k\right)App\left(i_{n_s}^k\right)\right) \tag{8}$$

where, n_s is the node with the minimum distance with respect to the actual position (x_t), $p(s_k)$ and $p(i_k)$ are the laser scan and image likelihood of *k* positions around the n_s, the *HF()* and *App()* are the Hausdorff Fraction and two-view geometry operators representing the respective appearance processes.

3. Experimental Results

We tested our approach on a Pioneer 3DX mobile robot equipped with an onboard computer at 1.5 GHz, an omnidirectional vision setup composed of a RemoteReality parabolic mirror with a diameter of 74mm, a UI-2230SE-C camera with a resolution of 1024x768 pixels, and a URG-04LX laser range finder. The omnidirectional vision system was previously calibrated. Using these calibration parameters a binary mask was previously computed in order to remove the central texture information where the robot is placed, and the outer texture information from the mirror. In addition, a LUT (look-up-table) was computed in order to lift the omnidirectional image points to the equivalent spherical model. We assumed a planar motion of the mobile robot and its navigation used collision-free trajectories. We collected a data set along two seasons (autumn and winter) and three different times of day. These datasets were used to incrementally build an indoor map of the P-1 floor at the PIV building in the University of Girona. The loop closure hypotheses generation was also tested.

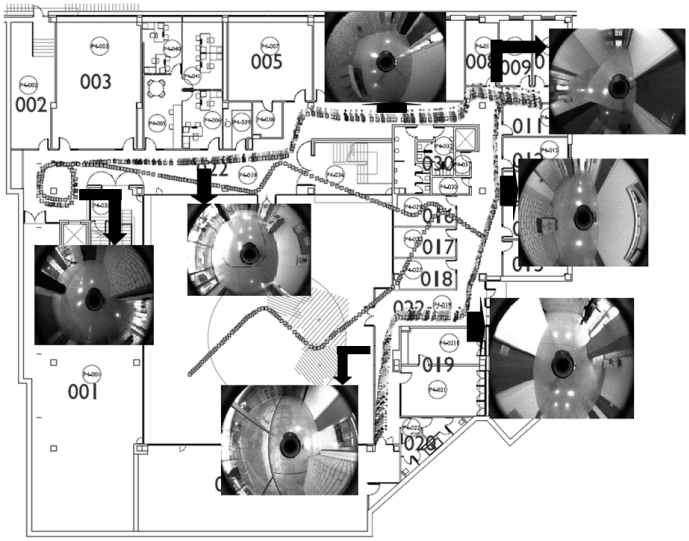


Figure 4. Robot trajectory obtained for the P-1 level at the PIV building in the University of Girona.

Figure 4 shows the resulting robot trajectory which was superimposed on the building plan. This figure shows the odometry readings and the robot positions obtained from the best particle along the robot trajectory. The robot trajectory was obtained using our probabilistic topological mapping approach, which combines the metric information provided by the laser scan as well as the appearance-based observations from the omnidirectional camera in order to build the topological map. Figure 4 shows how the uncertainty in the robot trajectory increases when the robot was approaching to the top-right part of the map, and then since the robot did a U-turn, the loop closure detection caused a reduction in the uncertainty. Figure 5 shows a zoom-in of the first part of the robot trajectory, which corresponds to the top-left region on the Figure 4. Figure 5 shows the particle distribution at each node in the first part of the robot trajectory. This figure shows also the odometry readings in order to compare them with the position estimates done by our approach. Figure 5 shows a set of small

circles which represent the particles dispersion centered in the position estimation of the best particle. The other particles are also shown, in some cases they are difficult to see given the black and white restrictions for the graphics. In some cases, the relative distance between nodes seems different; this is caused by Eq. (5) since from the appearance point of view (laser scans and image LTM-features) the intermediate observations were equivalent, i.e. the probabilistic appearance-based mapping approach proposed and implemented using a particle filter automatically performed a node reduction based on its appearance. However, the image features of these intermediate positions were used to update the local appearance of the environment.

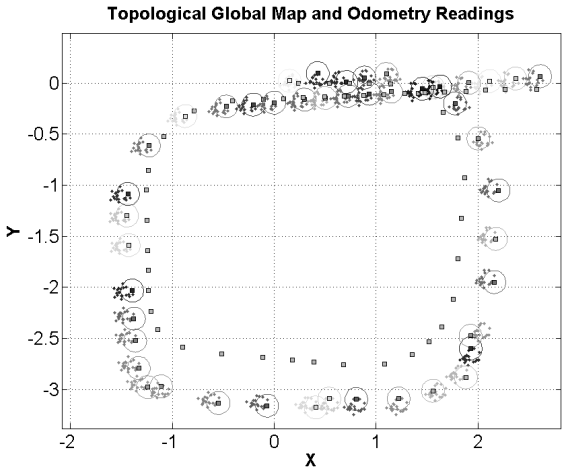
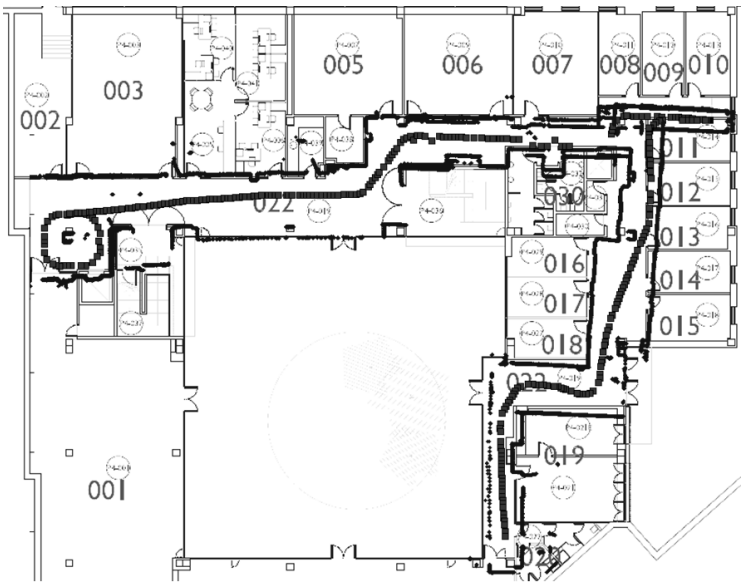
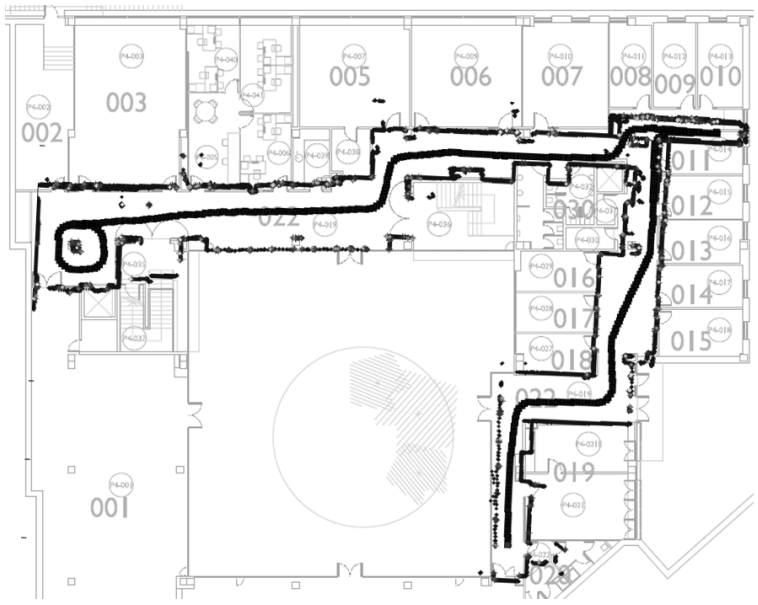


Figure 5. Detail of the particle distribution in the topological map as well as the odometry readings.



a)



b)

Figure 6. a) Global map before the loop closure detection and optimization using [17]. b) Global map after the loop closure detection and optimization using [17]. The robot trajectory is also shown.

The global map on the CAD of the P-1 floor obtained using our appearance-based mapping and localization approach is shown in Figure 6. According to the robot trajectory there are two possible loops, one in the top-left part of the map which had no impact on the environment topology due its early occurrence; and the second one in the top-right part of the map. Figure 6a shows the global map before the latter loop closure was detected and optimized using [17], and Figure 6b shows the global map after this optimization. Clearly, the global map created up to this moment changes specially the top-right part of the Figure 6a which is clearer in the Figure 6b, and the bottom-right part of Figure 6a which is not as straight as the same part of Figure 6b. Note that once the topological map was corrected our approach uses this improved version to incrementally grow up the map afterwards. The laser range finder used to validate our approach has a limited maximum range of 4m, which in open areas it gives no useful information about the environment and the odometry has to be used for the prediction step. Another situation where our system has difficulties is when the robot travels in corridors; there are not enough image descriptors, and the robot position depends only of the prediction step which is prune of error.

4. Conclusions

This work has improved our previous work in appearance-based mapping and localization presented in [12] using a Rao-Blackwellized particle filter framework to build a topological map and simultaneously locate the mobile robot. The approach presented uses the laser scan and omnidirectional images data to incrementally update

the particle weights and then the robot position on the topological map. The laser scan data were continuously compared using the Hausdorff Fraction metric and the image features were managed using the FSH approach, which together not only simultaneously build the map and locate the robot but they were used to generate loop closure hypotheses. These hypotheses were used to feed the TORO algorithm which was in charge of doing the respective optimizations. In future work we plan to improve the geometric uncertainty taking into account the appearance information, as well as to introduce image features closer to the appearance of the environment.

Acknowledgements

This work has been partially funded by the Commission of Science and Technology of Spain (CICYT) through the coordinated project DPI-2007-66796-C03-02, the LASPAU-COLCIENCIAS grant 136-2008, the University of Valle contract 644-19-04-95, and the consolidated research group's grant SGR2005-01008.

References

[1] J. Porta, B. Krose, Appearance-based concurrent map building and localization using a multi-hypotheses tracker, Intelligent Robots and Systems, IEEE/RSJ International Conference on, 4, (2004), 3424-3429.
[2] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool,. Omnidirectional Vision Based Topological Navigation. Int. J. Comput. Vision 74, 3, (2007), 219-236.
[3] Wongphati, M.; Niparnan, N.; Sudsang, A., Bearing only FastSLAM using vertical line information from an omnidirectional camera, Robotics and Biomimetics, IEEE Int. Conf. on , (2009), 1188-1193.
[4] K. L. Ho and P. Newman, Detecting loop closure with scene sequences, Int. Journal of Comp. Vision, vol. 74, no. 3, (2007), 261–286.
[5] Bazeille, S. and Filliat, D., Combining Odometry and Visual Loop-Closure Detection for Consistent Topo-Metrical Mapping, RAIRO Operations Research, (2010), 1-6.
[6] Botterill, Tom and Mills, Steven and Green, Richard, Bag-of-words-driven, single-camera simultaneous localization and mapping, Journal of Field Robotics, vol. 28, num. 2, (2011), 204—226.
[7] Mark Cummins and Paul Newman, Highly Scalable Appearance-Only SLAM – FAB-MAP 2.0, In Robotics Science and Systems (RSS). Seattle, USA. June (2009), 1-8.
[8] König, A., Kessler, J., Gross, H.-M., Improvements for an appearance-based SLAM-Approach for large-scale environments, Proc. 4th European Conf. on Mobile Robots (2009), 235-240.
[9] Ranganathan, A. & Dellaert, F., Online Probabilistic Topological Mapping,. The International Journal of Robotics Research, first published online before print January 24, (2011), 1-17.
[10] Montijano, E.; Sagues, C. ,Topological maps based on graphs of planar regions, Intel. Robots and Systems, IEEE/RSJ Int. Conf. on , (2009), 1661-1666.
[11] Gallegos, G.; Meilland, M.; Rives, P.; Comport, A.I., Appearance-based SLAM relying on a hybrid laser/omnidirectional sensor, Intl. Robots and Systems, IEEE/RSJ Int. conf. on , (2010), pp.3005-3010
[12] B. Bacca, J. Salvi, J. Batlle, X. Cufí, Appearance-based mapping and localisation using feature stability histograms, Electronics Letters , 46, 16, (2010), 1120-1121.
[13] R. Atkinson, R. Shiffrin. Human memory: A proposed system and its control processes, In K.W. Spence & J.T. Spence, The Psychology of Learning and Motivation, Academic Press, (1968), 89–195.
[14] Alan Baddeley. Working memory: Looking back and looking forward. Nat Rev Neurosci., 4, 10, (2003), 829-839.
[15] R. Llinás., I of the Vortex: From Neurons to Self, MIT Press, Cambridge, MA, 2001.
[16] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, SURF: Speeded Up Robust Features, Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, (2008), 346—359.
[17] Grisetti, G.; Lodi Rizzini, D.; Stachniss, C.; Olson, E.; Burgard, W., Online constraint network optimization for efficient maximum likelihood map learning, Robotics and Automation, IEEE Int. Conf. (2008), 1880-1885.
[18] Gutmann, J.-S. Konolige, K., Incremental mapping of large cyclic environments, Comp. Intell. in Robotics and Automation, Proceedings. IEEE Int. Symposium on, (1999), 318-325.

Towards an Efficient Use of Resources in All-Optical Networks[1]

Ramón BÉJAR [2], Cèsar FERNÁNDEZ, Francesc GUITART and Carles MATEU
Dept. of Computer Science, Universitat de Lleida, Spain

Abstract. Designing networks which can provide more and more bandwidth is a daunting and continuous effort. All-optical networks are one of the most successful recent approaches to tackle with that need although they come with some inconveniences. One of such problems is the need to design networks that will be able to cope with existing and future demands with the least possible hardware deployment, especially without having to resort to costly frequency conversion or opto-electronic conversion.

In this work we deal with this problem, named RWA-SLE, by encoding it as a pseudo-Boolean satisfiability problem. Then we compare results using our solving method with other proposed approaches for a wide range of generated problem instances. Results show that, for those problems where it is hard to find a suitable set of routes and wavelength assignments our method performs better than other methods. Solving those hard instances is particularly interesting because, otherwise, more hardware deployment would be needed to meet the traffic requirements.

Keywords. Satisfiability benchmarks, RWA, optical networks.

1. Introduction

Last years have seen a surge in all-optical network deployment that has come together with a dramatic increase in available bandwidth thanks to the use of Wavelength Division Multiplexing (WDM) on such networks.

These kind of networks work by allocating direct connections (circuits) between users, usually customers, traversing all the network. As the main idea is to use all-optical networks, such circuits must provide light continuity, and given that possible paths or routes are a finite resource, and that for every connection between network nodes only a limited set of wavelengths (or lambdas) are available, there is much interest in devising methods and algorithms to efficiently allocate routes and lambdas to each required connection (such pairs, route and lambda, are known as a lightpath).

This problem is known as RWA (Routing and Wavelength Assignment) problem. It can be solved in three flavours: as a static problem, knowing in advance all traffic demands we must attend (known also as RWA-SLE, Static Lightpath Establishment); as

[1]The research supported by the Spanish CICYT Projects ARINF (TIN2009-14704-C03-01), TASSAT (TIN2010-20967-C04-03) and University of Lleida pre-doctoral program.

[2]Corresponding Author: Department of Computer Science, Universitat de Lleida. Jaume II, 69. 25001, Lleida. SPAIN. E-mail: ramon@diei.udl.cat

an incremental problem, where new demands can appear at any time, but once established they stay so for ever; or as a dynamic problem, where demands appear spaced in time and must be attended as they appear, and after a working period they disappear and resources can be reclaimed for reuse. RWA has been proved to be NP-complete [6], thus making it an interesting research problem from the optimization point of view.

This work presents a new approach to encode the previously mentioned problem as a pseudo-Boolean satisfiability problem. Our new approach is then compared to a previously published SAT encoding approach [12][3] and to greedy algorithms traditionally used in networking literature. There is also recent work about the problem of maximizing the number of traffic demands that can be attended with a given network using constraint programming [10]. Experimental results provided show clearly that our approach outperforms other methods in instances where those methods have poor performance or are unable to provide a solution. Those cases used to be scenarios with a critical level of resources and with none or a few possible solutions.

2. Definitions

In this section we present some formal definitions for the problem we study in this paper. The routing and wavelength assignment problem (RWA) without wavelength translation is a generic term. In our case we focus on the RWA-SLE (Static Lightpath Establishment) without wavelength conversion on nodes. Future work will deal also with wavelength conversion, that is, RWA-T (RWA-Translation).

Every instance of the RWA-SLE problem is characterized by the following elements:

- Connection network $N = (V, E)$, where V is the set of nodes and E contains an undirected link $l_{i,j}$ for every pair of nodes i and j such that there is a link between them. To simplify the problem we consider that the capacity, different number of wavelengths a link can support, of all links is the same, denoted by parameter λ.
- Set of traffic demands R, where $r_{s,t} \in R$ if there is a demand for a lightpath between s and t.[4] In RWA without wavelength conversion, the wavelength must be the same for every link corresponding to the same route $r_{s,t}$, while in RWA-T it is assumed that a node can translate a wavelength to any other.[5]

3. Encodings

Once the problem is defined, we describe two main encoding approaches. First, we encode RWA using pseudo-Boolean formulation (PB). A first pseudo-Boolean encoding is described in [2]. We extend it by adding more powerful clauses and by proposing a more compact formulation that performs better for large problems. Second, we use a SAT for-

[3] We compare with this approach as it seems to be the "best" existing SAT approach.

[4] If we allow multiple lightpaths between each possible pair of nodes, then R is a multiset instead of a set. This can happen in "real-world" networks, as each lightpath can correspond to a different customer or application.

[5] In fact, usually not all nodes can convert wavelengths, and those that can are able only of a limited range and number of translations.

mulation proposed in [12], applying it to a broader range of problems and comparing it with our PB approach.

It is worth noting that a PB encoding allows to easily extend the formulation to the RWA-T problem, as for example, translating the linear programming approach of [11] to PB. Such an extension in SAT is a much harder task. Note that the solution for RWA-T in [12] is only valid for full-capable conversion nodes and cannot be generalized to network scenarios where the conversion range of the nodes is limited, as in real world networks.

3.1. Pseudoboolean Encoding

Our encoding is based on a set of propositional variables and constraints specified with pseudo-Boolean formulas. Some of the pseudo-Boolean constraints have a direct translation to logical clauses, but some of them have a more compact form as the pseudo-Boolean constraints we present.

For every route demand $r_{s,t} \in R$ we have the following propositional variables to encode all the possible routes in the communication network (V, E):

1. For every node $i \in V$, variable $n^i_{r_{s,t}}$ indicates if node i appears in the route for $r_{s,t}$.
2. For every link $l_{i,j}$, variable $d^{i,j}_{s,t}$ indicates if link $l_{i,j}$ appears in the route for $r_{s,t}$.
3. For every link $l_{i,j}$ and wavelength w, variable $x^{i,j,w}_{s,t}$ indicates if link $l_{i,j}$ appears in the route for $r_{s,t}$ using wavelength w.

So, the total number of propositional variables is $|R| \cdot (|V| + |E| \cdot (\lambda + 1))$.

The set of constraints can be divided in two groups, one that ensures the selection of a connected path for each route demand $r_{s,t}$, and another that avoids using the same wavelength for routes sharing a link. For the first group, we have the following pseudo-Boolean constraints:

- The starting and ending nodes of the route must be active:

$$n^s_{r_{s,t}} = n^t_{r_{s,t}} = 1,\ \forall r_{s,t} \in R \tag{p1}$$

- On a route, there is only one active link connected with s and t:

$$\sum_{l_{s,i} \in E} d^{s,i}_{s,t} = 1,\ \forall r_{s,t} \in R \tag{p2}$$

$$\sum_{l_{i,t} \in E} d^{i,t}_{s,t} = 1,\ \forall r_{s,t} \in R \tag{p3}$$

- For any node k, different from s and t, either it is active together with two adjacent links or it is not active and none of its links are active:

$$-2n^k_{r_{s,t}} + \sum_{k \neq \{s,t\}, j \neq \{s,t\}} d^{k,j}_{s,t} = 0,\ \forall r_{s,t} \in R \tag{p4}$$

For the second group, we have these constraints:

- If a link $l_{i,j}$ is active, then exactly one wavelength is used in that link, otherwise no wavelength is used:

$$-d_{s,t}^{i,j} + \sum_{w} x_{s,t}^{i,j,w} = 0,\ \forall r_{s,t} \in R,\ \forall l_{i,j} \in E \tag{w1}$$

- For any intermediate node (j) in a route, if a wavelength w is used in link $l_{i,j}$ for a given route, then it must also be used in adjacent link $l_{j,k}$ that is active in the same route:

$$-x_{s,t}^{i,j,w} - d_{s,t}^{j,k} + x_{s,t}^{j,k,w} > -2, \quad \forall r_{s,t} \in R, \quad \forall (l_{i,j}, l_{j,k}) \in E^2 | j \neq s,t; i \neq k \tag{w2}$$

  Observe that assuring the use of the same wavelength between adjacent links of the route is enough to ascertain the use of only on wavelength in all the links of the route.
- The same wavelength w cannot be used in a given link $l_{i,j}$ for different routes:

$$-x_{s,t}^{i,j,w} - x_{s',t'}^{i,j,w} > -2,\ \forall (r_{s,t}, r_{s',t'}) \in R^2 | \{s,t\} \neq \{s',t'\} \quad \forall l_{i,j} \in E,\ \forall w \tag{w3}$$

Related to this second group, two redundant or alternative constraints may be considered.

- For any route, if the starting active link $l_{s,i}$ uses wavelength w, then any other different link $l_{i',j'}$ in the route cannot use a different wavelength w' (and analogously for the ending active link $l_{i,t}$):

$$-x_{s,t}^{s,i,w} - x_{s,t}^{i',j',w'} > -2,\ \forall r_{s,t} \in R, \quad \forall (l_{s,i}, l_{i',j'}) \in E^2,\ \forall w \neq w'$$
$$-x_{s,t}^{i,t,w} - x_{s,t}^{i',j',w'} > -2,\ \forall r_{s,t} \in R, \quad \forall (l_{i,t}, l_{i',j'}) \in E^2,\ \forall w \neq w' \tag{w2b}$$

  This constraint achieves the same effect that w2 because for a connected path it is enough to ensure the use of a same wavelength between adjacent links.
- A wavelength w is used by at most one of the route demands in a link:

$$\sum_{r_{s,t} \in R} x_{s,t}^{i,j,w} \leq 1,\ \forall l_{i,j} \in E,\ \forall w \tag{w3b}$$

  This constraint achieves the same effect as constraint w3.

Given the complementarity between constraints w2 and w2b, and between w3 and w3b, we can consider four different PB encodings: PB_{2+3}, $\mathrm{PB}_{2+3\mathrm{b}}$, $\mathrm{PB}_{2\mathrm{b}+3}$, and $\mathrm{PB}_{2\mathrm{b}+3\mathrm{b}}$ obtained by selecting one from each pair of complementary constraints. It is worth noticing that in a solution of the previous constraints although for any route $r_{s,t}$ a connected path must be present, it can also contain some other activated links that will form isolated cycles. However, these cycles do not change the soundness of the encoded path between s and t.

3.2. A More Compact Pseudoboolean Encoding

The previous PB encoding considers individual variables for encoding the wavelength used in each link of a route, with the idea to be able to extend it to an encoding for the general RWA-T problem, where some nodes may be able to convert wavelengths. In case we are only interested in working with the non conversion variant of RWA, we can reduce the number of variables and constraints as we show in the following modified PB encoding (called compact PB encoding, PB_{cpt}).

The number of variables and constraints is reduced by considering only one variable $x^w_{s,t}$ per traffic demand and wavelength, denoting that the same wavelength w is used in all the links of the route between s and t. This new encoding substitutes the previous second group of constraints (set w) by the following:

- One wavelength must to be used per demand:

$$\sum_w x^w_{s,t} = 1,\ \forall r_{s,t} \in R \tag{cw1}$$

- For each pair of traffic demands sharing a link, at most one wavelength variable must be set:

$$-d^{i,j}_{s,t} - d^{i,j}_{s',t'} - x^w_{s,t} - x^w_{s',t'} > -4,$$
$$\forall (r_{s,t}, r_{s',t'}) \in R^2 | \{s,t\} \neq \{s',t'\},\ \forall l_{i,j} \in E,\ \forall w \tag{cw2}$$

3.3. Comparison with SAT Encoding

In [12] is presented a SAT encoding for RWA, i.e., based on a set of Boolean variables plus a CNF formula for encoding the problem. They showed that their approach outperformed previous specialized approaches [8] for solving RWA. Since we noticed that for some constraints a more compact pseudo-Boolean formulation was possible, and given the current state-of-the-art in efficient translations of pseudo-Boolean constraints to CNF formulas, it seemed natural to study possible improvements of their encoding by using a pseudo-Boolean encoding that was more compact and efficient, at least for some cases. That is the main reason for studying the encoding we present in this paper.

For each combination of node, route and wavelength, the SAT encoding of [12] uses a Boolean variable for indicating whether that node is active for that route using that wavelength. Analogously, for each combination of link, route and wavelength, it uses a Boolean variable for indicating whether that link is active. The fact that they do not use a specific variable for encoding whether a link is active for one route, independently of the wavelength used, is one of the reasons that increases the number of clauses of the problem with respect to the number of clauses in our encoding.

For the clauses used for encoding the constraints, we discuss only the ones that present significant differences with our encoding. When we consider the size of our PB constraints, we are assuming that we transform them to a CNF using the sorting network encoding of [7]. Recently, new CNF encodings for PB constraints have been proposed [4] that, although they have a bigger size, they promise to be more powerful, with respect

to propagation, in some cases. However, they still have to be successfully integrated in current SAT solvers.

To ensure end-to-end continuity for a route, the SAT encoding forces each node of the path to have the right number of active links in a different way. For ensuring that the start and ending nodes of a route have exactly one active link, it uses a set of clauses for ensuring *at least one link* plus a set of clauses for ensuring *at most one link*. This second set of clauses is the one that gives a significant difference in the number of clauses needed, compared with our corresponding PB constraints p2 and p3, as the number of clauses for this *at most one link* is $O((\lambda\ deg)^2)$ for the starting or ending node of a route but for our constraints p2 and p3 is $O(deg \log^2(deg))$. [6] We have an analogous situation for the constraint that ensures that the other nodes of the route have exactly two active links. The SAT encoding uses a set of clauses for ensuring *at most two active links per node*, that has a size of $O(deg^3\lambda^2)$, for each node of the route. By contrast, our corresponding PB constraint p4 has a size of $O(deg \log^2(deg))$ for each node of the route.

For ensuring that the same wavelength is used along all the links of a route, the SAT encoding uses a set of clauses that ensures that if a wavelength is used on the active link connected to the starting node of the route, then the other active links of the route cannot use other wavelengths, and analogously with the ending node. This set of clauses has size $O(|E|\lambda^2 deg)$ for each route demand. In contrast, we get the same effect with our sets of PB constraints w1 and w2 that have size $O(|E|\lambda(\log^2(\lambda) + deg))$. Actually, the set of clauses used by the SAT encoding is equivalent to the set of PB constraints defined in our alternative constraint w2b. For the $\mathrm{PB}_{\mathrm{cpt}}$ encoding the only constraint we need is cw1, that has size $O(|R|\lambda)$, smaller than in the other encodings.

To avoid using the same wavelength in a link in more than one route, the SAT encoding uses the set of clauses resulting from the transformation of the constraints defined in w3, so both have the same size: $O(|R|^2\lambda|E|)$. However, given that the alternative PB constraint w3b is a cardinality constraint[7], using the sorting network CNF encoding the total number of clauses will be $O(|R|\log^2(|R|)\lambda|E|)$. So, given that the level of consistency achieved by both w3 and w3b is the same, arc-consistency for the constraint: *a wavelength is used in any link by at most one route*, in the case of problems with a high number of route demands, using w3b instead of w3 can be an advantage. For the $\mathrm{PB}_{\mathrm{cpt}}$ encoding, constraint encoding cw2 has a bigger size than w3b: $O(|R|^2\lambda|E|)$. So, even if $\mathrm{PB}_{\mathrm{cpt}}$ is more compact with respect to the number of variables, this is done at the cost of increasing the size of the encoding for this constraint.

4. Experimental Results

To compare the performance of the above explained encodings we conduct experiments in two cases. First, by generating synthetic problems in a broad range of network scenarios by developing a problem generator and building a benchmark of instances. Second, by translating and encoding existing benchmarks based on real world networks. Although there exist some problem instances derived from those networks [8,12], we have

[6] deg is an upper bound on the number of links of any node.

[7] A pseudo-Boolean constraint in which all the coefficients are 1.

	Satisfiable						Unsatisfiable					
Time(s.)	2+3	2b+3	2+3b	2b+3b	cpt	SAT	2+3	2b+3	2+3b	2b+3b	cpt	SAT
< 1	9	1	32	5	0	14	10	9	55	25	31	19
$(1, 10]$	42	3	161	2	1	124	10	3	180	16	10	15
$(10, 10^2]$	3	2	2	0	17	29	1	1	4	1	5	0
$(10^2, 10^3]$	0	0	0	0	2	2	2	2	2	1	1	0
$\geq 10^4$	0	0	0	0	0	0	0	2	0	1	8	0
$\sum$	54	6	195	7	20	169	23	17	241	44	55	34

Table 1. Number of satisfiable and unsatisfiable instances solved faster per encoding at their corresponding time range.

found that they are too easy for all the solving approaches we compare in this section, probably because these instances are defined with a very high level of resources.[8]

4.1. A RWA problem generator

Our generator works by first creating a fully connected network and then generates a set of random traffic demands to be satisfied by the network. Problem instances for our experimentation are created using Waxman model [13] that creates a network topology that obeys a power law. Waxman model works by first placing nodes randomly across a bidimensional space, as is the case with real networks. With nodes placed then it proceeds to add edges (optical connections) between each pair of nodes u and v, using the probability function, $P(u, v) = \beta^{\frac{-d(u,v)}{L\alpha}}$ where $d(u, v)$ is the distance on the plane between u and v, L is the maximum distance between two nodes, and α and β are parameters in the range $(0, 1]$. Larger values of β represent higher connectivity degrees for the nodes, whilst α indicates connectivity from a node to more distant nodes, i.e. long-haul edges. For each edge we must also define the number of available wavelengths (λ).

Once the network is built, we generate a set of traffic demands D between random pairs of nodes. As the network is fully connected there is always a path between every possible pair of nodes but, as satisfying a demand uses up an available wavelengthi, there is no guarantee that all possible sets of demands will be satisfiable.

In order to evaluate the encodings performance, we have used a test set consisting of 980 instances, for network topologies from 10 to 20 nodes, from 20 to 50 traffic demands, and 2 to 10 available lambdas, creating a wide range of problems from unsatisfiable to easily satisfiable.

4.2. Hardness characterization

In RWA problems, as in other studied problems [9,1,3], also exist an easy-hard-easy hardness characterization when we move from unsatisfiable to satisfiable instances along a given parameter, the number of lambdas in our case. As an example, Table 3 shows the percentage of satisfiable instances and the computation time to solve 20 instances of a problem with 10 nodes, 20 traffic demands, $\alpha = 0.65$ and $\beta = 1$, for different values of λ. As we focus on network sustainability, as traffic demands grow, we must be able to

[8] Due to space constraints, detailed results for real world networks are not included, although authors have them available on request.

study and solve the RWA problem under regimes at which the number of resources is at a critical level.

With these goals in mind, we have designed our problem instance generator to obtain a wide range of problem instances. From instances with a high number of solutions, to instances with no solutions, passing trough an intermediate class of instances in which the level of resources is at a critical level. Such instances have very few solutions and are very hard to solve, especially with greedy algorithms. With our generator, these different classes of instances are easily generated by simply modifying the parameter λ as we keep fixed the other parameters.

4.3. Encodings Performance

Table 1 summarizes the performance of our four different basic PB encodings (PB_{2+3}, $\mathrm{PB}_{2+3\mathrm{b}}$, $\mathrm{PB}_{2\mathrm{b}+3}$, and $\mathrm{PB}_{2\mathrm{b}+3\mathrm{b}}$), our compact PB encoding ($\mathrm{PB}_{\mathrm{cpt}}$) and the SAT encoding (SAT).[9] The results are presented separated for instances with solution (left table) and for instances with no solution (right table). Each column shows the number of instances that the corresponding encoding has solved faster. The timeout to stop the SAT solver, with any encoding, has been 23 hours per instance. Even with such timeout, some instances have not been solved with some of the encodings.

For satisfiable instances, the encoding $\mathrm{PB}_{2+3\mathrm{b}}$ has the best performance for the easiest instances, but as the difficulty increases, the SAT and $\mathrm{PB}_{\mathrm{cpt}}$ encodings perform better. Overall, we observe that the encoding $\mathrm{PB}_{2+3\mathrm{b}}$ does better although followed closely by the SAT encoding. The $\mathrm{PB}_{\mathrm{cpt}}$ encoding only gives an advantage for some difficult satisfiable instances.

For unsatisfiable instances, that represent cases with not enough resources to satisfy all traffic demands, we again observe that $\mathrm{PB}_{2+3\mathrm{b}}$ performs well for easy instances, but when difficulty increases $\mathrm{PB}_{\mathrm{cpt}}$ improves. We believe that this is due to the more compact form of representing assignment of wavelengths to routes in $\mathrm{PB}_{\mathrm{cpt}}$, such that there are no partial solutions with more than one wavelength assigned to the same path of a route. This fact may allow the SAT solver to reach more quickly the branches of the search tree that represent different assignments of wavelengths to the routes. However, given the higher size of the constraint that avoids using the same wavelength in different conflicting routes in $\mathrm{PB}_{\mathrm{cpt}}$, $O(\lambda|E||R|^2)$, with respect to the one of $\mathrm{PB}_{2+3\mathrm{b}}$, $O(\lambda|E||R|\log^2(|R|))$, the possible improvement thanks to a more compact search space may be reduced due to the increased time needed to check the constraints.

To compare the relative performance of our best encodings, $\mathrm{PB}_{2+3\mathrm{b}}$ and $\mathrm{PB}_{\mathrm{cpt}}$, with the SAT encoding, in a more quantitative way, we have also created scatter plots, where every point (x, y) in a plot represents one of the instances of the test-set with the x value representing its solving time with one encoding and the y value its solving time with a second encoding. Figure 1 shows a scatter plot between the SAT and the $\mathrm{PB}_{\mathrm{cpt}}$ encodings (left plot) and a scatter plot between the SAT and the $\mathrm{PB}_{2+3\mathrm{b}}$ encodings (right plot). We observe that $\mathrm{PB}_{\mathrm{cpt}}$ is either competitive or superior (especailly for unsatisfiable instances) to the SAT encoding. $\mathrm{PB}_{2+3\mathrm{b}}$ seems to be clearly superior to the SAT encoding on unsatisfiable instances, but for satisfiable instances, the SAT encoding is almost always the best. However, the experiments at the end of this section show that as the

[9] The solver used for SAT solving, as well as solving CNF encoding of PB encodings is Precosat [5]. The CNF formulas are generated from the corresponding PB encodings using Minisat+ using sorters [7].

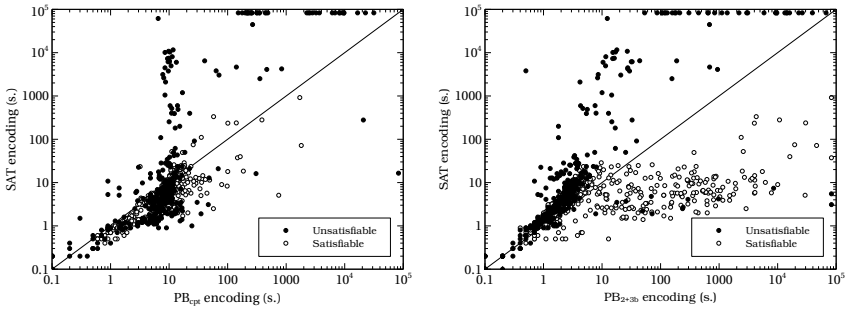


Figure 1. Scatter plot of times to solve satisfiable and unsatisfiable instances for different encodings.

	Time(s.)			
# demands	$\mathrm{PB_{cpt}}$	$\mathrm{PB_{2+3b}}$	SAT	NEW
25	10	8	-	1
50	-	20	-	u
75	-	123	-	u
100	-	1,395	-	u
125	-	-	-	u

Table 2. Time in seconds to solve an instance for 20 nodes, $a = 0.4$, $b = 0.5$, and $\lambda = 20$. (-) indicates memory exhaustion, and (u) unsatisfiable reported.

λ	% sat	Mean time (s.)
2	40	0.59
3	60	0.79
4	70	192.39
5	90	1.58
6	90	2.18

Table 3. An example of easy-hard-easy hardness characterization on RWA problems.

number of demands increases, even for satisfiable instances, $\mathrm{PB_{2+3b}}$ starts to outperform the SAT encoding.

We have also tested the performance of the best solving approach reported in [14], named NEW algorithm in that reference, that can be seen as an improved version of the common approach used to solve the RWA problem in two phases with a greedy approach. First, a transformed graph is used to determine the shortest route between source and destination using the Dijkstra shortest path algorithm, where the weights of the links quantify the probability for a request to pass through a node, and later a simple greedy scheme is used to assign a proper wavelength to that path. Once a wavelength is assigned, the weights of the routing graph are updated, in order to attend future demands. Although it is an adaptive algorithm that tries to reduce the blocking probability, its greedy nature makes it very inefficient for RWA-SLE problem because, when considering the combined problem (routing + wavelength assignment), it is possible that it is unable to find a solution for a problem with a certain number of wavelengths, although the problem is perfectly solvable.

Our tests show that for a given network and demand set, our approach needs less wavelengths to satisfy them. As an example, for the same network with 10 nodes reported above in Table 3, the NEW algorithm only finds satisfiable instances for λ above 10, and even for $\lambda = 10$, the percentage of satisfiable found instances is a scarce 10%. As greedy algorithms perform better for a large λ, we conduce some additional tests for a larger λ and more traffic demands. Concretely, we take 20 nodes, $a = 0.4$, $b = 0.5$, and $\lambda = 20$. Table 2 shows the time to solve an instance for different values of traffic demands. Note that all the instances are satisfiable.

From Table 2 one can derive some facts. First, NEW algorithm is not able to find solutions when demands go beyond 25. Greedy algorithms do not do particularly good when resources are scarce. Second, due to the encoding size, the SAT encoding is not able to solve even the smaller case, 25 demands, because of memory exhaustion (1 GB of RAM). The best performing PB encoding is $\mathrm{PB}_{2+3\mathrm{b}}$, which is able to solve up to 100 traffic demands.

5. Conclusions

We studied the use of current SAT solvers for the resolution of the RWA-SLE problem. We proved that a good formulation allocates network resources more efficiently than other approaches, as greedy algorithms, extensively studied in the specialized literature, at least, for more critically constrained problems. We also contributed new PB encoding variants, highly competitive with the existing SAT formulation for this problem, that make easier the task of extending the proposed formulation to RWA problems with partial lambda translation. And finally, we reported a comparative performance test among our encodings, SAT encodings and greedy algorithms under different network scenarios.

References

[1] Dimitris Achlioptas, Carla Gomes, Henry Kautz, and Bart Selman. Generating satisfiable problem instances. In *Proceedings of the AAAI 2000*, pages 256–261. AAAI Press / The MIT Press, 2000.

[2] Fadi A. Aloul, Bashar Al Rawi, and Mokhtar Aboelaze. Routing in optical and non-optical networks using boolean satisfiability. *Journal of Communications*, 2(4):49–56, 2007.

[3] Carlos Ansótegui, Ramón Béjar, Cèsar Fernández, Carla Gomes, and Carles Mateu. Generating highly balanced sudoku problems as hard problems. *Journal of Heuristics*, pages 1–26, 2010. 10.1007/s10732-010-9146-y.

[4] Olivier Bailleux, Yacine Boufkhad, and Olivier Roussel. New encodings of pseudo-boolean constraints into cnf. In *SAT*, pages 181–194, 2009.

[5] Armin Biere. Precosat version 236. http://fmv.jku.at/precosat, 2009.

[6] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: an approach to high bandwidth optical wan's. *Communications, IEEE Transactions on*, 40(7):1171 –1182, July 1992.

[7] Niklas Eén and Niklas Sörensson. Translating pseudo-boolean constraints into SAT. *JSAT*, 2(1-4):1–26, 2006.

[8] Jeffery L. Kennington and Eli V. Olinick. Wavelength translation in wdm networks: Optimization models and solution procedures. *INFORMS Journal on Computing*, 16(2):174 –187, 2004.

[9] P. Prosser. An empirical study of phase transitions in binary constraint satisfaction problems. *AI Journal*, 81:81–109, 1996.

[10] Helmut Simonis. A hybrid constraint model for the routing and wavelength assignment problem. In *Proceedings of the 15th international conference on Principles and practice of constraint programming*, CP'09, pages 104–118, Berlin, Heidelberg, 2009. Springer-Verlag.

[11] P.N. Tran and U. Killat. An exact ILP formulation for optimal wavelength converter usage and placement in WDM networks. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1 –6, December 2008.

[12] J. Valavi, N. Saluja, and S.P. Khatri. A boolean satisfiability based solution to the routing and wavelength assignment problem in optical telecommunication networks. In *Communications, 2005. ICC 2005. 2005 IEEE International Conference on*, volume 3, pages 1802 – 1806 Vol. 3, May 2005.

[13] B.M. Waxman. Routing of multipoint connections. *Selected Areas in Communications, IEEE Journal on*, 6(9):1617 –1622, December 1988.

[14] Yongbing Zhang, Koji Taira, Hideaki Takagi, and Sajal K. Das. Adaptative wavelength routing and assignment in optical wdm networks. *Optical Networks Magazine*, 3(5):86–99, 2003.

Depth of Valleys Accumulation Algorithm for Object Detection

Jorge Bernal, Javier Sánchez and Fernando Vilariño
Computer Vision Center, Campus Universitat Autònoma de Barcelona, Edifici O, 08193 Bellaterra, Barcelona, Spain

Abstract. This work aims at detecting in which regions the objects in the image are by using information about the intensity of valleys, which appear to surround objects in images where the source of light is in the line of direction than the camera. We present our depth of valleys accumulation method, which consists of two stages: first, the definition of the depth of valleys image which combines the output of a ridges and valleys detector with the morphological gradient to measure how deep is a point inside a valley and second, an algorithm that denotes points of the image as interior to objects those which are inside complete or incomplete boundaries in the depth of valleys image. To evaluate the performance of our method we have tested it on several application domains. Our results on object region identification are promising, specially in the field of polyp detection in colonoscopy videos, and we also show its applicability in different areas.

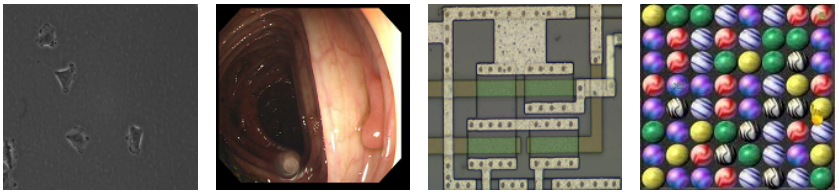
Keywords. Object Recognition, Object Region Identification, Image Analysis, Image Processing

1. Introduction

Object recognition is a recurring challenge in computer vision and artificial intelligence, concerned on answering questions such as what objects are in the picture and where? Currently it remains challenging mainly because the high variations that real-world images present. We could see object recognition schema as a three parts scheme: object detection, object description and object classification. Object detection methods find out where objects are in the image, no matter what they are. Object recognition algorithms must overcome some of the problems such as partial occlusions, viewpoint changes or variation in the illumination. Some of the available solutions in the context of pattern recognition need of the use of an image segmentation procedure followed by a machine learning approach to perform the final classification.

In this paper we present a novel object detection method that, without any image segmentation, detects which region of the image is more likely to contain an object inside by using valleys information. Valleys surround objects in images where the camera and the light source are in the same (or opposite) direction, which happens in microscopic, X-ray or endoscopic images, just to name a few (see Figure 1).

Our first contribution extends the definition of the depth of valleys image [3], which uses as an approximation that the boundaries that surround objects in image can be seen as intensity valleys. The problem with those intensity valleys is that in many cases they



(a) (b) (c) (d)

Figure 1. Example images (a) cells (b) endoscopic image (c) integrated circuit board (d) marbles.

are not closed and, therefore, we can not know if they are defining the boundary of an object or they are just non-important boundaries. Our method uses an approximation denoted as the depth of valleys image that not only uses the information about where valleys are but also their intensity so the points that belong to the boundaries of objects have a high value in the depth of valley image and the points interior to objects have a lower value in the depth of valleys image.

Our second contribution, ellipse fitting depth of valleys accumulation (EF-DOVA) consists of finding which points on the image belong to objects and which do not, by checking if they are surrounded, in a high number of directions by points with high value in the depth of valleys image.

The results of our method can be used for several applications that need of an object detection, as a way to indicate which parts or region of the image are more likely to contain objects. In this paper we show how our method can be applied in several real-world scenarios such as cell detection and polyp detection in colonoscopy videos. Both contributions are meant to work together in order to build a complete object detection algorithm but the accumulation algorithm, as we will show for cell detection, could also be applied to images where a edges or blob detection is more suitable (Figure 1 (d)).

The structure of the paper is as it follows: in Section 2 we present the ideas extracted from other papers that inspired our work. In Section 3 we present our algorithm. In Section 4 we present results of the applications of our method. In Section 5 we analyse the results obtained and compare them with other methods. Finally in Section 6 we expose the main conclusions along with future research lines.

2. Related work

In this paper we present an object detection method that detects automatically where objects appear in the image. This work can be enclosed into the category of regions of interest detectors, which are generally capable of reproducing similar performance that the human would provide in locating elemental features in images.

Feature detection is often a low-level image processing operation, usually performed as the first operation on an image, and examines every pixel to see if there is a feature present at that pixel. There are several feature detection methods that can be divided into four main groups: edge detectors (with methods such as Canny or Sobel [4]), corner detectors (such as Harris [5] or SUSAN [13]), blob detectors (such as Laplacian of

Gaussians [10], or MSER [9]) and region of interest detectors (SURF [2] or Intensity Extrema-based Region Detector [10]). In our case we take advantage of the presence of valleys in certain types of images, in order to develop our object detection method.

As it was mentioned in Section 1, object detection is often the first step in an object recognition method so, once the regions of interest are located in the image, we need to describe them in a way such the posterior object classification can give good results. There are many types of descriptors, usually divided according to the type of feature we are interested, such as shape, color or texture.

After the description of the detected objects is done, object classification is performed. It usually involves a learning procedure to find out which combination of features describe better an object so, when a new object arrives into the processing pipeline, it can be classified accurately. As it can be thought, having a three parts schema implies that the output of an stage (in this case the object detection stage) affects the results of the following stage hence the important of a good object detection, which is the objective of the work presented in this paper.

3. Methodology

3.1. Depth of valleys

As mentioned in Section 1, objects tend to appear as enclosed by intensity valleys when the source of light and the camera share the same or opposite direction (see Figure 2).

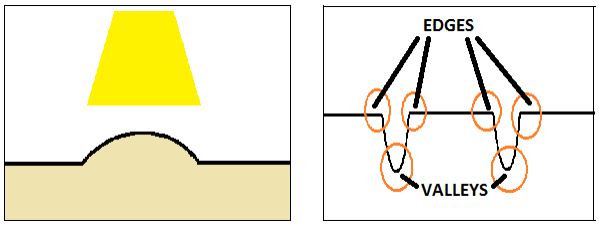


Figure 2. Simulation of (a) an illuminated prominent surface (b) grey scale profile [3].

As it is explained in [8], ridges and valleys in bi-dimensional images are commonly identified as loci of minimum gradient magnitude along the relief's level curves. In 2D, ridges/valleys can be also identified as positive maxima/ negative minima of the curvature of the relief's level curves. Maxima are connected from one level to the next therefore constituting a subset of the vertex curves. As it is shown in Figure 3 the output of a ridges and valleys detector applied in one of our target images informs us about where valleys are in the image but not about their intensity. The results of these detectors are also affected by the presence of reflections or artifacts in the image.

In order to obtain information about the intensity of the valleys morphological gradient can be used. It is defined as the difference between the dilation and the erosion of

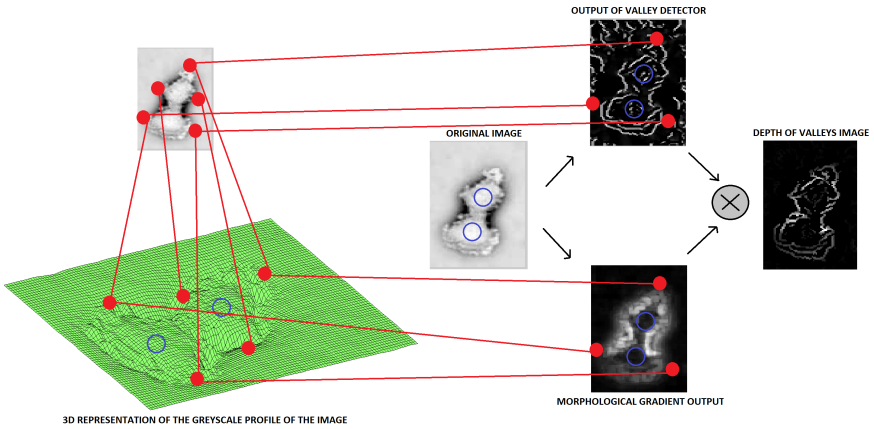


Figure 3. Example of the calculation of a depth of valleys image.

a given image [12] and gives as output an image where each pixel value indicates the contrast in intensity in the close neighborhood of that pixel.

As it can be seen in Figure 3, the combination of the information obtained from the ridges and valleys detector (V) with the one that the morphological gradient (MG) provides (see equation 1) gives as a result what we could call as the *depth of valleys image* (DV). In the points where we have a valley and the morphological gradient value is high (red points in Figure 3), we will have a high value of the deep of valleys image but in the points where we have a valley but the morphological gradient is slow, the depth of valley will be minima (blue circles in Figure 3).

$$\forall i, j \in \mathbf{I} \quad DV(i,j) = V(i,j) \cdot MG(i,j) \tag{1}$$

We can see in Figure 3 how the depth of valleys has higher values in the points that constitute the 'real' valleys of the image and lower to points inside the valley.

It is important to mention that the ridges and valley extractor that we have used [8] needs of two parameter values (σ_d: differentiation scale and σ_i: integration scale) that must be put in correspondence with the size of the structural element (sd) that we use (in our case, a disk) to calculate the morphological gradient. More precisely, σ_i and sd should be equal in order to work in the same scale. If this does not happen, maxima points of the ridges and valleys extractor can be located in places where the morphological gradient is not maxima and therefore the desirable properties of the resulting depth of valleys image would be lost.

The next step consist of finding which parts of the image constitute the individual objects on the image, which will be separated by boundaries constituted by points with high values in the depth of valleys image.

3.2. Ellipse fitting depth of valleys accumulation algorithm

EF-DOVA algorithm consist of five different steps:

1. *Choice of the starting point:* The possibilities are either to use as starting points the minima of the depth of valleys image or the minima of the ridges and valleys image. Once

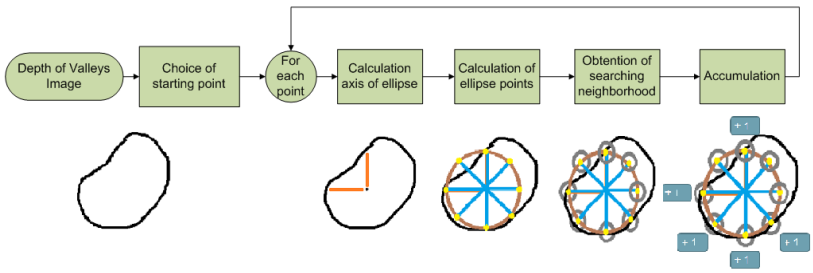


Figure 4. Graphical explanation of the depth of valleys accumulation algorithm.

the algorithm knows which are those points it starts the process. We have approximated the objects to be detected as having ideally an elliptical shape (although they do not appear as perfect ellipses in the majority of images) and our approach consists of trying to find the points in the image that fit better depth of valleys ellipses.

2. *Calculation of the axis of the ellipse:* For each starting point we find the maxima of depth of valleys image in both x and y direction. The positions of these maxima will constitute the axis of the ellipse that we want to approximate. Although the explanation is based in axis-centered ellipse, EF-DOVA algorithm also considers rotated ellipses.

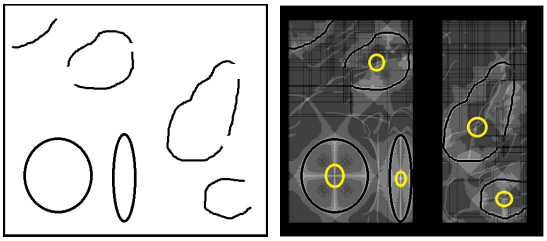
3. *Calculation of ellipse points:* In order to denote a point as center of the ellipse that has been approximated, it has to coincide with the depth of valleys image in a certain number of directions (that we can define as a threshold). In this case we consider 8 directions that go from $\Theta = 0$ to $\Theta = 360$ separated by 45 each point from another. In order to calculate the rest of the points we equal the equation of our approximated ellipse with the equation of the line with slope the tangent of each angle (eq. 2).

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1; \quad y = \tan\Theta \cdot x; x = \frac{a \cdot b}{\sqrt{a^2 \cdot m^2 + b^2}}; \; y = \tan\Theta \cdot \frac{a \cdot b}{\sqrt{a^2 \cdot m^2 + b^2}} \quad (2)$$

4. *Obtention of the searching neighborhood:* Once we have these 8 points we define a neighborhood around them where search for maxima in the depth of valleys image.

5. *Accumulation* To calculate the accumulation value for the point we count how many of the 8 ellipse points have maxima in their neighborhood.

The way the algorithm works can be better understood by observing in Figure 4 how it would work on an ideal binary depth of valleys image. Accumulation values go from 2 to 8. We will denote as high accumulation values those higher or equal than 5. In Figure 5 we can see how our algorithm works on a synthetic image that contains some of the shapes that we may find (with both closed and unclosed contours), some of them with even not a clear elliptical shape.



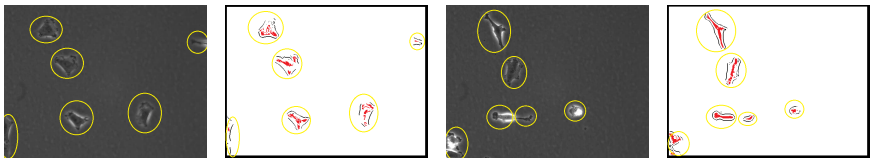
(a) (b)

Figure 5. (a) Original image (b) Accumulated values image.

4. Experimental results

4.1. Cell detection

With the increasing availability of live cell imaging technology, tracking cells and other moving objects in live cell videos has become a major challenge for bioinformatics [11]. One problem inherent for most tracking algorithms is over or under segmentation of cells, meaning that they tend to recognize one cell as several cells or vice-versa. Our approach is different than the one presented in [11] because they segment to identify cells (using flux tensors for detection of moving objects) while we do not need to do any segmentation to detect the different objects that appear in the image. As we show in Figure 6 we are able to detect which the cells are by only using our accumulation method, even when the intensity valley profile is far from closed. As it can be seen, we detect where cells are, even in cases where cells have just been divided (see Figure 6 (d).



(a) (b) (c) (d)

Figure 6. Example of cell identification (a,c) original images (b,d) binarized depth of valleys images (in black) with maxima of DOVA (in red).

We have measured, for the 361 frames of the video, the number of cells correctly detected and the number of false positives (objects detected that are not cells). In order to measure the accuracy of our test we have used the F_1 score, where $F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$. This measure considers both the precision and recall of the test to compute the score. The accuracy is better as the F_1 score is nearer to 1, meaning that, in the context of cell detection, we identify correctly a high number of regions that contain cells and in a very reduced number of cases we denote as a cell region one that does not

contain a cell. In Figure 7 we show the results of F_1 score for our method and how they are affected by the scale parameter (value of σ_i and sd). As it can be seen, results are very good when we take values around 13 for the scale parameter.

Scale parameter	4	7	9	11	13	15
F1 score	0.839	0.878	0.88	0.893	0.898	0.8965

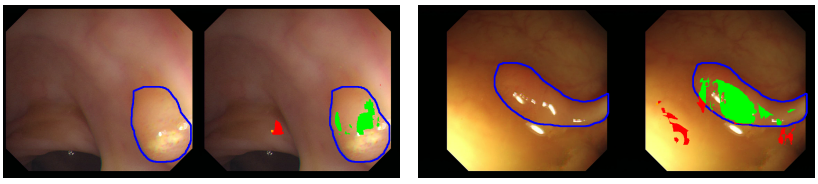
Figure 7. Results of F_1 score with respect to the scale parameter.

4.2. Polyp detection in colonoscopy videos

We have also tested our method in polyp detection in colonoscopy videos. Colonoscopy [6] is an screening technique used to detect colon cancer, which survival rate decreases as later it is detected hence the importance of an early detection. Many approaches have been carried in order to detect polyps in colonoscopy videos [1]. One of the main problem associated with this type of images is the great variability in polyp appearance (going from flat to peduncular shapes, or the view, from lateral to overhead) which makes unfeasible common template matching algorithms, among others.

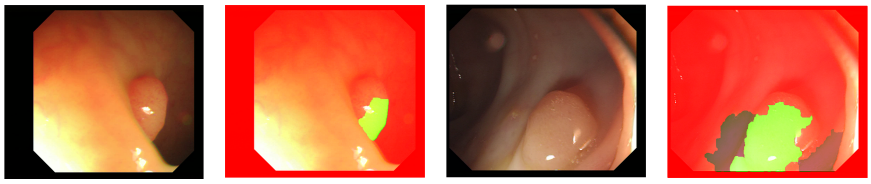
We show here how our method can help on reducing the part of the image where search for polyps. As it can be seen in Figure 8, our method locates maxima of accumulated value inside the polyp region (which are marked in green), no matter the type of polyp or the point of view of the image. It is true that it also places maxima in parts where is no polyp (marked in red), but our method offer an interesting result: if there is no maxima of accumulated value in some part of the image there will be no polyp in there, constituting a necessary but not sufficient condition of polyp presence.

In order to see how our method could help in polyp detection, we have colored areas in image according to the rate of pixels with high accumulated value rate in the area out of the total of points with high accumulated value, painting in reddish colors regions with low rate and in greenish colors those which high rate. The results (which can be seen in Figure 9 show that, although the type of polyp appearance in the images is different, our method colors in strong green parts of the image where the polyp really is. These results could be improved by using a more specific segmentation scheme.



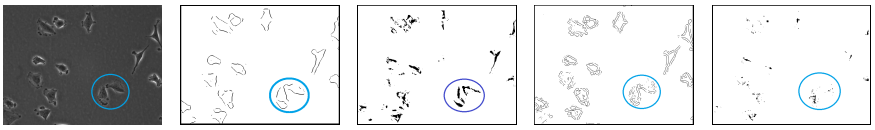
(a) (b)

Figure 8. Example of the apparition of maxima of accumulated value inside polyp regions.



(a) (b) (c) (d)

Figure 9. DOVA applied on polyp detection (a and c) original images (b and d) colored images.



(a) (b) (c) (d) (e)

Figure 10. Qualitative comparison between the results of DOVA and laplacian of gaussian (a) original image (b) binarized depth of valleys image (c) DOVA image (d) laplacian of gaussian image (e) accumulation of laplacian of gaussian.

5. Discussion

As shown in Section 4, our method offers good results in object region identification in several scenarios. The results could be improved if we adapt our method to specific problems (in the case of cell detection, we could study cues that indicate when cells are dividing) or we consider more kind of shapes to be approximated (although as of now, our method is able to cope with incomplete boundaries of any shape (see Figure 6).

We have compared the performance of our method with other approaches for the case of cell detection, where we have compared the results obtained with our complete cell detection algorithm with the results obtained by the combination of the Laplacian of Gaussians (LoG [10]) and our accumulation algorithm, to show the effect of using the DV image to obtain the boundaries of the objects (which can be seen as blobs as a whole) and to present how our accumulation algorithm can be used together with another feature detectors. In Figure 10 we show qualitative results of this comparison. For the sake of the comparison, we have compared the best result of our method with the best results achieved with LoG in terms of the objects' boundaries and applied the same parameters for the accumulation method (using LoG's output as the DV image). We can see that there are less maxima of accumulated value inside the cells for the case of LoG that, in some cases (see Figure 10 (d) may lead to lose the object.

In order to do a quantitative comparison we have measured, for frame of the video, the following: 1) percentage of the cells that have been correctly detected (cd; 2) number of false positives (parts of the image where we detects really non-existing objects); 3) percentage of pixels with high accumulated values near the center of the object; 4) the

	Experiment 1	Experiment 2	Experiment 3	Experiment 4	Experiment 5
DOVA	81.57%	12	26.55%	3.95	2.79
LoG + Accumulation	80.06%	3	22.84%	4.28	2.65

Table 1. Summary of the results. Experiment 1: percentage of correctly detected cells; Experiment 2: False positives; Experiment 3: Rate of high accumulation points near objects' center; Experiment 4: Mean accumulated value and Experiment 5: Ratio of high accumulation points versus low accumulation points per object.

mean accumulated value per object and 5) the ratio between pixels with high accumulated value vs pixels with zero as accumulated value per object. Results are shown in Table 1.

We can see the effect that has the use of DV image over LoG in object detection. In terms of object detection by applying our complete detection method, we detect more cells than when we apply the accumulation algorithm after LoG (81.57% against 80.06%, 30 cells) but, in the other hand, we present a higher total number of false positives (although the number is low). The reason why we do not have a perfect cell detection is that, in some cases of cell division, we detect one cell where there are really two. We can also see from Experiment 3 that by using EF-DOVA we obtain more pixels with high accumulated value near the real center of the object. In terms of mean accumulated value the results are very similar and we can see, from Experiment 5, that the ratio between the number of high accumulated values out of the total of pixels in the object is higher for DOVA. The main conclusion that can be extracted from this experiment is that the use of the DV image lead not only to a better cell detection but the number of pixels with high accumulated value near the center of the object is higher with our method, so we locate better the real center of the object.

The results of our method could be used, for example, to guide segmentation using high accumulation points as markers or, as a first step in an object recognition scheme where, once found which regions of the image contain objects, the next step often consists of describing them by using feature descriptors. The problem that we want to solve will tell us which type of feature descriptor we want to test (shape, texture or color).

The final objective in object recognition tasks is to classify the objects that appear in the image. To do so there are two main philosophies: generative (such as principal component analysis [14]) and discriminative methods (like support vector machines [7]). Generative approaches approximate original data to keep as much original information as possible whereas discriminative approaches are designed having classification tasks in mind. Classification goal is, given training data, to find the optimal decision criteria. In a generative model the likelihood of a sample is estimated and then assigned the most likely class while in discriminative models a label is assigned directly based on the decision criteria. As a classifier needs good inputs in order to perform well, we think the focus should be on object detection and description stages, in order to build accurate classification systems.

6. Conclusions and future work

In this paper a novel approach to object detection by object region identification has been presented, which takes advantage of the appearance of valleys surrounding objects. We have presented our two contributions: the definition of the depth of valleys image and

the accumulation algorithm. Both are meant to work together although the accumulation algorithm could be applied, as it has been shown, to any kind of boundary image.

We have applied our method in several scenarios, showing promising results for different experiments (such as cell or polyp detection). In cell detection we show that our method is able to detect a great percentage of the cells that appear across all the images. In the case of polyp detection we show that by applying our method we offer a necessary but not sufficient condition of polyp presence.

Although our method could be improved if we adapt it to solve specific problems we have shown the strength and versatility of our method, even compared with other methods. Another future research line could be to consider other shapes in the approximation, such as squares or a more specific choice of the starting point in the accumulation. The results shown in this paper indicate that, considering all the possible modifications and improvements, our object detection method could be a powerful tool in object recognition (especially in images with the valley presence is clear).

ACKNOWLEDGEMENTS

This work was supported in part by a research grant from Universitat Autonòma de Barcelona 471-01-3/08, by the Spanish Government through the founded project "COLON-QA" (TIN2009-10435) and by research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018).

References

[1] S. Ameling, S. Wirth, and D. Paulus. *Methods for Polyp Detection in Colonoscopy Videos: A Review*. Inst. fur Computervisualistik, 2009.

[2] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. *Computer Vision–ECCV 2006*, pages 404–417, 2006.

[3] Bernal, J., Sánchez, J., and Vilariño, F. A Region Segmentation Method for Colonoscopy Images Using a Model of Polyp Appearance. In *Proceedings of the, IbPRIA 2011*, page (in press), Las Palmas de Gran Canaria, Spain, June 2011.

[4] R. Fisher, S. Perkins, and A. W. et al. Sobel operator.

[5] C. Harris and M. Stephens. A combined corner and edge detection. In *In Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[6] Hassinger, J.P., Holubar, S.D. et al. Effectiveness of a Multimedia-Based Educational Intervention for Improving Colon Cancer Literacy in Screening Colonoscopy Patients. *Diseases of the Colon & Rectum*, 53(9):1301, 2010.

[7] M. Hearst, S. Dumais, and E. e. a. Osman. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28, 2002.

[8] López, A.M., Lumbreras, F. et al. Evaluation of methods for ridge and valley detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):327–335, 1999.

[9] J. Matas, O. Chum, M. Urbana, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.

[10] K. Mikolajczyk, T. Tuytelaars, and C. S. et al. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1):43–72, 2005.

[11] A. Mosig, S. Jager, and C. e. a. Wang. Tracking cells in Life Cell Imaging videos using topological alignments. *Algorithms for Molecular Biology*, 4(1):10, 2009.

[12] Serra, J. *Image Analysis and Mathematical Morphology*. Academic Press, Inc. Orlando, FL, USA, 1983.

[13] S. M. Smith and J. M. Brady. Susan - a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.

[14] F. Wood, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometr. Intel. Lab. Syst*, 2:37–52, 1987.

Extending Recommendation Systems with Semantics and Context-Awareness: Pre-Filtering Algorithms

Victor CODINA[a,1] and Luigi CECCARONI[b]

[a] *Departament de Llenguatges i Sistemes Informàtics (LSI), Universitat Politècnica de Catalunya (UPC), Campus Nord, Edif. Omega, C. Jordi Girona, 1-3, 08034 Barcelona, Spain*

[b] *Barcelona Digital Technology Centre (BDigital) Roc Boronat 117, 5a planta, 08018 Barcelona*

Abstract. During the last decade, several recommendation systems have been proposed that help people to tackle information overload of digital content by effectively presenting content adapted to user's tastes and needs. However, these personalization technologies are far from perfect and much research is needed to improve the quality of recommendations and, particularly, user satisfaction. In this paper we analyze and extend two relatively recent approaches for improving the effectiveness of recommendation systems: context-aware recommenders, which mainly focus on incorporating contextual information to the recommendation process; and semantically-enhanced recommenders, which focus on incorporating domain semantics. Although these approaches are compatible, how to properly combine them to maximize their strengths is still an unexplored research issue. The objective of this work is to provide the basis for this research. Concretely, we propose and evaluate an improved content-based model that exploits semantics and contextual information in an integrated way.

Keywords. Personalization, User Modeling, Knowledge-based Recommender Systems.

Introduction

In the mid-nineties, *recommendation systems* (RSs) or *recommenders* emerge as an independent research field of *information retrieval* and *artificial intelligence* to address the information overload problem by using a specific type of information filtering techniques that attempts to recommend information items (e.g., movies, TV programs/shows/episodes, videos on demand, music, books, news, images, Web pages, research papers) that are likely to be of interest to the user.

In the literature [1], [2], [3], *traditional recommendation strategies* are classified on the basis of their *knowledge source* or of the *algorithmic technique employed* (see section 1.1). When classified on the basis of the knowledge source, the following

[1] Corresponding author: vcodina@lsi.upc.edu

strategies can be identified: *content-based strategies* (CB), which express user's tastes as keyword-based user profiles and consist of recommending items matching user's preferences and item's features; and *collaborative filtering* (CF), which expresses user preferences as item ratings and whose recommendations are based on matching users or items with similar rating.

Although current recommendation systems achieve in general a good performance in terms of prediction accuracy, they still require additional improvements to obtain this performance in all application domains. Particularly, the main limitations identified in current recommendation strategies are: *lack of context-awareness*, *lack of transparency* (in explaining how the recommendations are generated), the *cold-start* problem (incapability of producing acceptable recommendations in initial situations where there is little user feedback) and data *sparsity*. So far, these limitations have been addressed independently, and no recommendation approach has been proposed that solves them effectively in an integrative way. In the following, two relatively recent approaches for improving the effectiveness of recommendation systems are analyzed: context-aware recommenders and semantically-enhanced recommenders. These approaches are the basis for the approach proposed in this paper.

Context-aware recommendation systems (CARSs) address the issues related with the production of contextual recommendations (see section 1.2). Some recent works in the topic (e.g., [4]) demonstrate that, depending on the application domain and the available data, contextual information is useful for providing more accurate recommendations. However, this approach suffers from the same limitations as traditional RSs, such as *sparsity* and *cold-start* problems, since it is a generalization of them. Actually, often these limitations are aggravated, because most CARSs follow a reduction-based approach [5], in which only user data associated with the active context is used for recommendation.

Semantically-enhanced* (or semantics-based) *recommendation systems (SERSs) are a relatively recent approach for overcoming the limitations due to lack of transparency, the cold-start problem and data sparsity through the incorporation of domain semantics. This approach assumes that these limitations are strongly related with the lack of understanding (and exploitation) of domain semantics of traditional recommenders, and relies on ontology-based representations of knowledge to take advantage of this domain semantics.

Taking into account that each recommendation approach is useful to overcome part of the limitations mentioned above, it is plausible to foresee that combining these two approaches (i.e. incorporating both semantics and contextual information) will provide better results. To date, little work has been carried out to study this kind of hybrid systems and additional research is needed to understand under which conditions they can outperform current recommenders. The objective of this work is to provide the basis for this research by proposing different semantically-enhanced, context-aware recommendation algorithms that incorporate semantics and contextual information into well-known recommendation strategies such as CB and CF.

The remainder of this paper is organized as follows. Section 1 presents a short review of the main traditional recommendation approaches as well as the two recent approaches that extend them in order to overcome some of their limitations. Section 2 describes semantically-enhanced CARSs, including the representation of entities of the recommendation space and a semantically-enhanced, context-aware CB model. Section 3 presents the experimental evaluation of the proposed model. Finally, section 4 presents conclusions and future work.

1. Background and Related Work

1.1. *Traditional Recommendation Systems*

Traditional recommenders try to model the degree of utility of the item i for the user u as a (real-valued) function $R(u,i)$ for the (user, item) pairs in which items have not been rated yet by the users. Based on the kind of algorithmic technique employed for the estimation of function $R(u,i)$, two main RS categories (both for CB and CF) can be distinguished [6]: *heuristic-based* (also called memory-based) RSs, which employ some heuristic formula, such as vector-based similarity and correlation measures, to calculate the recommendation; and *model-based* RSs, which generate the recommendation using a model learnt by applying some model-building technique over the user-item rating matrix. Heuristic-based systems adapt better to changes in user's preferences but scale worse than model-based ones.

Typically, *heuristic-based CB* [7] are inspired by *information retrieval* methods and calculate the *user-item* matching based on vector measures such as the cosine similarity. *Model-based CB* [8] calculate the matching based on a model of user's tastes built by applying *machine learning* techniques, such as *naïve Bayesian networks*, to text categorization.

Heuristic-based CF employs heuristic techniques such as rating correlation analysis, and is also known as neighborhood-based approach (based on the k-nearest-neighbor algorithm). Depending on if a subset of users or items is chosen to compute recommendations, two classes of heuristic-based CF can be identified: *user-based* ones [9], when the algorithm focus on finding users similar to the active one for recommending; and *item-based* ones [10], when the algorithm focus on finding items similar to those the active user likes. *Model-based CF* uses the ratings to learn a compact predictive model to represent *user-item* interactions.

1.2. *Context-Aware Recommendation Systems*

Unlike traditional recommenders, CARSs estimate the degree of utility of the item i for the user u as a function of not only items and users, but also context, i.e. $R(u,i,c)$. Context is a multifaceted concept used across various disciplines, each one considering it from its own perspective. According to Dourish [11], context can be defined as: a *representational view*, a predefined set of observable attributes, the structure of which does not change significantly over time; and an *interactional view*, which assumes the user behavior is influenced by an underlying context, but that the context itself is not necessarily observable. Most CARSs in the literature follow the representational view of context; some authors (e.g., [4]) adopt the interactional view, and model context through a short-term memory approach borrowed from psychology.

CARSs can be broadly categorized into two groups: (1) those using *context-driven querying and search*, where contextual information is used to search a certain repository of resources [12]; and (2) those using *contextual preference elicitation and estimation*, where traditional techniques are extended with context-awareness by modeling context-sensitive tastes. In this paper we focus on CARSs based on contextual preference elicitation and estimation, which can be further classified into three different paradigms according to how contextual information is incorporated in the recommendation process [13]: contextual pre-filtering, contextual post-filtering and contextual modeling.

In CARSs with *contextual pre-filtering*, context is used for selecting the relevant set of user data. Then, recommendation can be generated using any traditional recommendation algorithm on the selected data. The most common pre-filtering approach is the *reduction-based* one [5], which builds a local estimation model that uses only the ratings pertaining to the context in which a recommendation is made.

In *contextual post-filtering* approaches, contextual information is ignored when generating recommendations and incorporated to adjust the resulting predictions to the user context. The adjustments can be made by filtering out items of a recommended list that are irrelevant, or adjusting a rating prediction according to the context.

In *contextual modeling* approaches, contextual information is used as an explicit predictor of $R(u, i, c)$, the relevance estimation function. These approaches use multidimensional algorithms, while contextual pre-filtering and post-filtering approaches can use traditional algorithms. The only works following the contextual modeling paradigm found in the literature are based on generalizations of model-based approaches [14], [15].

1.3. *Semantically-Enhanced Recommendation Systems*

Unlike traditional recommenders, SERSs exploit items' domain knowledge. SERSs are a specific kind of hybrid recommenders [16] that combines knowledge-based (KB) methods with CB and CF, and, in particular, with heuristic-based recommendation algorithms, which are especially suitable to exploit semantics.

Two main types of reasoning are employed in SERSs: *semantic inference* methods, mainly used to infer implicit user's tastes from the explicit ones in order to make user profiles less sparse, therefore reducing the cold-start problem and data sparsity; and *semantic similarity* measures, used for the semantic matching between entities of the recommendation space in order to obtain more accurate similarity judgments than traditional similarity measures.

Semantically-enhanced CB recommenders are hybrids that incorporate KB methods to extend CB with semantics, following in most cases a feature augmentation strategy. Domains where this approach is commonly applied are those whose items are mainly represented as textual features extracted both from: unstructured data, such as news articles [17], [18], and research papers [19]; and structured data, such as features of TV programs [20] and of artworks [21]. The recommendation process of semantically-enhanced CB can be divided into three major steps: (1) semantic analysis of item's content, in which the relevant features of the items previously rated by the user are mapped to concepts of the domain ontology; (2) *ontology-based user modeling*, in which user's tastes are learnt/inferred based on the concepts identified in the previous step; (3) prediction generation, in which an estimation of function $R(u, i)$ is calculated by measuring the *user-item* semantic matching.

Semantically-enhanced CF incorporates KB methods to extend neighborhood-based CF. Mobasher et al. [22] extend item-based CF, but extension of user-based CF is more common [23], [24], [25]. Semantically-enhanced CF can be divided into three steps: (1) *ontology-based user modeling*, in which the tastes of the user and those of her neighbors are modeled as in semantically-enhanced CB; (2) *neighborhood formation*, where the k-nearest-neighbors are selected by measuring the semantic matching between user models (*user-user* matching); (3) prediction generation, in which function $R(u, i)$ is estimated by using a traditional user-based rating prediction method.

2. Semantically-Enhanced CARSs

To further improve the effectiveness of recommendations and overcome major limitations, this work aims to extend RSs by incorporating domain and contextual knowledge. In this paper, by *domain knowledge* we mean an ontology-based (or equivalent) representation that describes the features of the items to recommend (e.g., a classification of book genres) and by *contextual knowledge* we mean an ontology-based (or equivalent) set of attributes that are observable by the system (e.g., time of the day or location). We present a method to extend *heuristic-based CB* with semantics and context-awareness in an integrative way. However, the general idea of this method is also applicable, with some minor changes, to extend *heuristic-based CF*.

2.1. *Representation of Entities of the Recommendation Space*

In order to exploit the semantics defined in the ontology-based representations of domain and contextual knowledge, entities of the recommendation space should be represented as sets of concepts in the same semantic space. In CARSs, three types of entities form the recommendation space: users, items and context. We define each entity as follows:

- **User**. A user $u \in U$ is represented as a set of weighted concepts $P_u = (u_1, u_2, \dots, u_n)$, where each element u_x is the degree of interest in concept $d_x \in Domain(I)$ (i.e., each concept defined in the domain ontology, e.g.: *Action*). Here, concepts represent user's tastes or interest topics. Then, u_x could be the degree of interest of the user in action items (e.g., movies).
- **Item**. An item $i \in I$ is represented as a set of weighted concepts $P_i = (i_1, i_2, \dots, i_n)$, where each element i_x stores the relevance of concept $d_x \in Domain(I)$ in the description of the item. Here, concepts represent the features of the item. Then, i_x could be the relevance of "action" in item (e.g., movie).
- **Context**. A context state $c \in C$ is represented as a set of concepts $P_c = (c_1, c_2, \dots, c_n)$, where each element is a contextual attribute $c_x \in Domain(C_k)$. C_k corresponds to an observable type of contextual information (e.g., time of the day, weather, location, social situation). Then, in the case of "time of the day", c_x could be: morning, afternoon, evening.

2.2. *Semantically-Enhanced, Context-Aware CB*

Here, we extend *heuristic-based CB* with semantics and context-awareness, concretely, applying a *pre-filtering* contextual paradigm to semantically-enhanced CB. A global model is maintained in which contextual and temporal effects are captured as different biases that are independent of the user-item interaction (see [26] for a similar approach). In this way, no user data is neglected and data sparsity is not increased as in reduction-based approaches. The *prediction* rule of the proposed model is as follows:

$$\hat{r}_{ui}(c,t) = b_{ui} + Match(P_u, P_i) \quad (1)$$

where b_{ui} is the baseline prediction for the unknown rating in context c and timestamp t (see section 2.2.1 for the exact definition of baseline prediction); and $Match(P_u, P_i)$ is the function that estimates the *user-item* matching between the contextualized tastes of the user (P_u) and the features describing the item (P_i). Traditional CB use

$\sum_{d\in(P_i\cap P_u)} i_d \cdot u_d(c,t)$, as user-item matching function, where only the concepts appearing in both P_u and P_i are taken into account. In contrast, a semantic matching also exploits the relations between distinct concepts of P_u and P_i. To calculate the contextualized degree of interest, $\mathrm{u_d(c,t)}$, we consider both the long-term degree of interest u_d and the short-term effects (see also section 2.2.1):

$$u_d(c,t) = u_d + \alpha_{ud} \cdot |t - t_u|^{\beta} + b_{ud,t} + \sum_{c_x \in P_c} b_{ud,c_x} \quad (2)$$

In order to learn the parameters capturing the contextual and temporal effects we minimize the associated *root mean squared error* (RMSE) function on the training set using a regularized stochastic gradient descent optimization algorithm. The algorithm loops through all ratings in the training data. For each given rating r_{ui} a prediction ($\hat{r}_{ui}$) is made, and the associated prediction error $e_{ui} = r_{ui} - \hat{r}_{ui}$ is computed. To reduce the RMSE, for a given training case r_{ui}, each parameter p is modified in the opposite direction of the gradient, yielding $p \leftarrow p + \gamma \cdot (e_{ui} - \lambda \cdot p)$, where γ is the learning rate and λ the regularization meta-parameter, whose values are set by cross-validation.

2.2.1. Baseline Predictors

Baseline predictors (or biases) sum up effects, which do not depend by concrete user-item interactions, such as systematic tendencies for some users to give higher ratings than other users, or for some items to receive higher ratings than other items.

Following Koren [26], we start defining a static baseline (without temporal effects):

$$b_{ui} = \mu + b_u + b_i \quad (3)$$

where μ is the overall average rating, and b_u and b_i the observed deviations of user u and item i from the average. We then improve the definition of the baseline expressing it as a function of time:

$$b_{ui} = \mu + b_u(t) + b_i(t) \quad (4)$$

We finally extend the baseline predictor by adding the contextual effects:

$$b_{ui} = \mu + b_u(c,t) + b_i(c,t) \quad (5)$$

where, for the *user bias*, three effects are captured using the function $b_u(c,t) = b_u + \alpha_u \cdot |t - t_u|^{\beta} + b_{u,t} + \sum_{c_x \in P_c} b_{u,c_x}$, where: b_u is the stable part of the user bias; $\alpha_u \cdot |t - t_u|^{\beta}$ is a linear model for approximating a gradual drifting behavior, being t_u the user's mean date of rating, and $|t - t_u|$ the number of days between dates t and t_u; $b_{u,t}$ captures session-specific variability, modeling short-lived effects, such as a different mood of the user that day; b_{u,c_x} captures context-specific variability modeling user periodic patterns, such as different attitudes during the weekends. For the item bias two effects are captured: $b_i(c,t) = b_i + b_{i,Bin(t)} + \sum_{c_x \in P_c} b_{i,c_x}$, where b_i is the stable part of the item bias; $b_{i,Bin(t)}$ captures temporal variability modeled by time-based bins of the same size; and b_{i,c_x} captures context-specific variability modeling item's periodic patterns such as the popularity of the item in specific seasons of the year.

3. Experimental Results

3.1. *The MovieLens Dataset*

Experimental evaluation is carried out using the MovieLens movie dataset, which is the only publicly available dataset that provides numerical user's ratings along with extra information to obtain contextual and domain knowledge. We use rating timestamps and user's demographics to obtain the *temporal context* about each rating, such as the season and the part of the day in which the user rated the movie as well as whether it was during the weekend or on a weekday.

As *domain knowledge* we consider the dataset as a folksonomy. (A folksonomy describes users, resources, tags, and user-based assignment of tags to resources.) The dataset includes the movie genre classification as a predefined, formal tag. Other tags are obtained by natural language processing of user generated text included in the dataset. Examples of popular tags are: "will smith", "exciting", "cartoon", "very funny". Tags are used as topics of interest. Because the movie genre used to classify the movies consist of a plain list of general concepts, in the experiments we can only obtain domain semantics from user-generated tags. We automatically extract an explicit taxonomy of concepts from the folksonomy, by analyzing co-occurring terms assigned to items and determining the super- and sub-classes of terms based on conditional relative frequencies.

In order to compare the performance of the proposed CB algorithm to other non-CB algorithms, only concepts with a certain amount of content are considered: therefore we prune the original dataset by selecting only tags that are applied to at least five different movies and by three different users. Moreover we only select ratings of movies that contain at least five different tags. In Table 1 the dataset statistics before and after pruning are shown.

Table 1. Dataset statistics before and after pruning

	Before pruning	**After pruning**
Movie ratings	1,000,209	722,782
Users	6,040	5,604
Movies	3,883	1,243
User-based assignment of tags	95,580	64,023
Topics	16,529	1,815

3.2. *Experimental Design and Metrics*

We evaluate the proposed context-aware CB model by comparing the performance of four model variants of Eq. (1):

- *Static-CB* and *Contextual-CB*, which employ the non-semantic user-item matching along with the static baseline predictor defined in Eq. (3) and the contextual baseline defined in Eq. (5), respectively. Moreover, *Contextual-CB* uses the contextualized user modeling defined in Eq. (2).
- *Static-SemCB* and *Contextual-SemCB*, which consist of the *Static-CB* and *Contextual-CB* models extended with user-item semantic matching. The matching function we employ is an *all-pairs* method that counts the u_d of the parent topics of P_i that are contained in P_u in addition to the topics of P_i that perfectly match P_u.

In all model variants, the relevance of each item's topic (i_d) is measured as the tag frequency of the item when the topic is a user-generated tag, and as the genre popularity when the topic is a genre. We found that applying a sigmoid transformation improves the overall performance. The parameters of the gradient descent algorithm are set by cross-validation to $\gamma = 0.005$; $\lambda = 0.002$; $nºiterations = 5 - 10$.

In order to provide insight into algorithm performance, we compare the various model variants to five naïve baselines: *item-avg* (prediction only based on item's rating average); *itemUser-avg* (prediction based on item's rating average plus the difference between user's rating average and the overall average); *static-baseline*, defined in Eq. (3); *linear-baseline*, defined in Eq. (4); and *contextual-baseline* defined in Eq. (5). We compare results also with one of the best-performing *model-based CF*, the *singular value decomposition* (SVD) with the *expectation-maximization* (EM) procedure to deal with the sparsity of ratings (*SVD-EM)*, which approximates the full user-item rating matrix to a lower-dimension matrix of K factors (we set $K = 50$).

All the algorithms were evaluated using five-fold cross validation and two evaluation metrics: *recall at ten* (top-10) for evaluating *top-N recommendation* (typical values for N range between 5 and 20), and *root mean square error* (RMSE), for evaluating *rating-prediction* precision. For the top-10 evaluation we create a test set containing the last 5-star-rated item of each test user. Then for each test user we form a top-10 recommendation using as possible relevant items all the unrated items (plus the last 5-star-rated item) by this user. If the test item is in the top-10 list then we have a hit, otherwise we have a miss. Finally, we compute the recall as the number of hits divided by the number of test items. For the *rating prediction* experiment we create a test set containing the last six rated item of each test user, following the method used in the KDD 2011 cup contest[2]; the rest of the ratings were used for training and validating the recommendation algorithms. In addition to the global error we measured the cold-start performance of each model using the eight percent of users with fewer ratings, corresponding to less than 25 movies in this experiment.

3.3. *Results and Discussion*

Table 2 shows the experimental results for the different recommendation algorithms. Higher recall values and lower RMSE values correspond to better performance. We carried out significance tests at $p < 0.05$ in order to analyze in more detail the differences between algorithms.

From the analysis of the results we can draw the following conclusions: *linear-baseline* and *contextual-baseline* slightly reduce global RMSE with respect to the *static-baseline*, but differences between them are not significant; CB models extended with context-awareness (*contextual-CB* and *contextual-SemCB*) reduce the RMSE with respect to the *static-CB*; models extended with semantics (*static-SemCB* and *contextual-SemCB*) slightly reduce the RMSE with respect to non-contextual models, above all in cold-start users, although the improvement is not very significant; differences between naïve baselines and the all CB model variants are not significant in terms of top-10 performance.

Although the improvements achieved in this experiment by extending the CB model with semantics and context-awareness are relatively small, it has to be taken into

[2] http://kddcup.yahoo.com, accessed 7 July 2011.

account that these results are influenced by the characteristics of the dataset (e.g., the average number of sessions per user is approximately three and the number of ratings per session is high, which implies that contextual effects and gradual drift behavior are less relevant than session-specific effects, in this experiment). Results are also influenced by the limited semantics extracted from the folksonomy. (Only 92 hierarchical relationships were inferred from 1797 topics.)

Table 2. Experimental results

	Recommendation Algorithm	RMSE		Recall (TOP-10)	
		Global	Cold-Start	Global	Cold-Start
Naïve Baselines	item-avg	0.99 ± 0.03	1.02 ± 0.03	0.031 ± 0.002	0.03 ± 0.02
	itemUser-avg	0.92 ± 0.03	0.99 ± 0.03	0.031 ± 0.002	0.03 ± 0.02
	static-baseline	0.89 ± 0.02	0.95 ± 0.03	0.030 ± 0.003	0.03 ± 0.02
	linear-baseline	0.88 ± 0.02	0.96 ± 0.02	0.028 ± 0.005	0.03 ± 0.02
	contextual-baseline	0.88 ± 0.02	0.96 ± 0.02	0.029 ± 0.005	0.02 ± 0.02
Model variants	static-CB	0.85 ± 0.02	0.92 ± 0.02	0.034 ± 0.003	0.03 ± 0.02
	static-SemCB	0.84 ± 0.02	0.92 ± 0.02	0.036 ± 0.003	0.04 ± 0.02
	contextual-CB	0.84 ± 0.02	0.92 ± 0.02	0.035 ± 0.005	0.03 ± 0.02
	contextual-SemCB	0.84 ± 0.02	0.92 ± 0.02	0.034 ± 0.005	0.03 ± 0.02
CF	SVD-EM	0.82 ± 0.02	0.92 ± 0.02	0.064 ± 0.011	0.05 ± 0.02

4. Conclusions and Future Work

In this paper we analyzed and extended two relatively recent approaches for improving the effectiveness of recommendation systems: context-aware recommenders, which mainly focus on incorporating contextual information to the recommendation process; and semantically-enhanced recommenders, which focus on incorporating domain semantics. Concretely, we have proposed and evaluated *semantically-enhanced, context-aware, content-based strategies applying a pre-filtering contextual paradigm*, which outperform traditional heuristic-based, content-based strategies in terms of prediction accuracy. The results show that the improved content-based strategies are not far in terms of performance from the results of one of the best *model-based collaborative filtering strategies*. Moreover, the proposed recommender overcomes the lack of transparency of model-based approaches because predictions are based on explicit topics of the domain.

As future work we will experiment with new ways of integrating context-awareness and semantics to traditional *heuristic-based collaborative filtering*. An approach we are currently experimenting with is to employ semantic inference methods in order to infer user's tastes that are related with those learnt from user feedback. We think that this type of inferences may be useful to improve the performance of the recommenders, especially in cold-start conditions.

References

[1] Goldberg, D., Nichols, D., Oki, B. and Terry, D. Using collaborative filtering to weave an information tapestry. *Communications of the Association of Computing Machinery*, 35(12):61-70, 1992.

[2] Melville, P. and Sindhwani, V. Recommender Systems. In C. Sammut and G. Webb (Eds.) *Encyclopedia of Machine Learning*, 829-837. Springer, 2010.

[3] Ricci, F., Rokach, L., Shapira, B. and Kantor, P. B. (Eds.) *Recommender Systems Handbook*, 842 pages. Springer, 2011.
[4] Anand, S. and Mobasher, B. Contextual recommendation. In B. Berendt et al. (Eds.) *WebMine 2006, LNAI* 4737:142–160, 2007.
[5] Adomavicius, G., Sankaranarayanan, R., Sen, S. and Tuzhilin, A. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems*, 23(1):103–145, 2005.
[6] Breese, J., Heckerman, D. and Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th conference on Uncertainty in Artificial Intelligence*, pages 43–52, San Francisco, 1998. Morgan Kaufmann.
[7] Ahn, J., Brusilovsky, P., Grady, J., He, D. and Syn, S. Open user profiles for adaptive news systems: help or harm?. In *Proceedings of the 16th international conference on World Wide Web*, pages 11–20, 2007.
[8] Billsus, D. and Pazzani, M. A hybrid user model for news story classification. In *Proceedings of the Seventh International Conference on User Modeling*, 99-108, June 1999.
[9] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L. and Riedl, J. GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77-87. 1997.
[10] Linden, G., Smith, B. and York, J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Computing*, 7(1):76-80. 2003.
[11] Dourish, P. What we talk about when we talk about context. *Personal and Ubiquitous Computing*, 8(1):19-30. 2004.
[12] Ardissono, L., Goy, A., Petrone, G., Segnan, M. and Torasso, P. Intrigue: Personalized Recommendation of Tourist Attractions for Desktop and Hand Held Devices. *Applied Artificial Intelligence*, 17(8):687-714, 2003.
[13] Adomavicius, G. and Tuzhilin, A. Context-aware recommender systems. In *Recommender Systems Handbook*, pages 335–336. 2010.
[14] Oku, K., Nakajima, S., Miyazaki, J. and Uemura, S. Context-aware SVM for context-dependent information recommendation. In *proceedings of 7th International Conference on Mobile Data Management*, page 109, 2006.
[15] Karatzoglou, A., Amatriain, X., Baltrunas, L. and Oliver, N. Multiverse Recommendation: N-dimensional Tensor Factorization for Context-aware Collaborative Filtering. In *proceedings of the fourth ACM conference on Recommender systems*, pages 79-86. 2010.
[16] Burke, R. Hybrid Web Recommender Systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl (eds), *Adaptive Web*, 4321:377-408. 2007
[17] Cantador, I., Bellogín, A. and Castells, P. Ontology-based personalised and context-aware recommendations of news items. In *proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, 1:562–565. 2008.
[18] Shoval, P., Maidel, V. and Shapira, B. An Ontology Content-Based filtering method. *International Journal on Information Theories and Applications*, 15:303-318. 2008.
[19] Middleton, S., Shadbolt, N. and De Roure, D. Ontological User Profiling in Recommender Systems." *ACM Transactions on Information Systems*, 22(1):54-88. 2004.
[20] Blanco, Y., Arias, J., Ramos, M. and Lopez, M. Providing entertainment by content-based filtering and semantic reasoning in intelligent recommender systems. *IEEE Transactions on Consumer Electronics*, 54(2):727-735. 2008.
[21] Wang, Y., Wang, S., Stash, N, Aroyo, L. and Schreiber, G. Enhancing Content-based Recommendation with the Task Model of Classification. *In proceedings of 17th International Conference EKAW 2010*, LNCS 6317:431-440, Lisbon, Portugal. 2010.
[22] Mobasher, B., Jin, X. and Zhou, Y. Semantically Enhanced Collaborative Filtering on the Web. In Berendt, et al. (eds.) *Web Mining: From Web to Semantic Web*. LNAI 3209:57-76. Springer. 2004.
[23] Degemmis, M., Lops, P. and Semeraro, G. A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation. *User Modeling and User-Adapted Interaction*, 17(3):217-255. 2007.
[24] Sieg, A., Mobasher, B. and Burke, R. Improving the Effectiveness of Collaborative Recommendation with Ontology-Based User Profiles. In *proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems*. 2010.
[25] Liu, P., Nie, G. and Chen, D. Exploiting semantic descriptions of products and user profiles for recommender systems. In *Computational Intelligence and Data Mining, IEEE Symposium*, 179–185. 2007.
[26] Koren, Y. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89-97. 2010

Negotiation Based Branch & Bound and the Negotiating Salesmen Problem

Dave DE JONGE and Carles SIERRA
Artificial Intelligence Research Institute, IIIA-CSIC
Campus de la Universitat Autònoma de Barcelona
{davedejonge,sierra}@iiia.csic.es

Abstract. We introduce a new multiagent negotiation algorithm that explores the space of joint plans of action: NB^3. Each negotiator generates a search tree by considering both actions performed by itself and actions performed by others. In order to test the algorithm we present a new variant of the Traveling Salesman Problem, in which there is not one, but many salesmen. The salesmen need to negotiate with each other in order to minimize the distances they have to cover. Finally we present the results of some tests we did with a simple implementation of the algorithm for this problem.

Keywords. Negotiation, Search, Negotiating Salesmen Problem

Introduction

Negotiation algorithms have frequently been used to co-ordinate autonomous agents. Although negotiation has rightly been described as a search problem [1], previously proposed negotiation algorithms have mostly focused on the utility space. These algorithms assume that given a utility aspiration level it is always possible to find a proposal that would fit that level. This is often not possible when the domains of the issues are discrete or when there are integrity constraints among them [2]. In this paper we focus on complex problems for which the classical continuity assumptions do not apply and thus solutions have to be found directly at domain level. Also, we address a number of realistic assumptions that make the application of current negotiation algorithms unfeasible:

- The space of solutions is huge, i.e. there is no possibility to exhaustively explore the set of solutions.
- Solutions improve with co-operation. Some actions are interdependent, if agents help each other they are individually better off.
- The environment is only partially observable by each agent, e.g. actions made by others may not be observable.
- The environment changes due to actions of others.
- Decisions have to be made within a limited time frame.
- Solutions may involve a large number of agents, possibly including humans.

Many difficult problems belong to this class, e.g: school time table construction [3], route scheduling for package delivery companies [4], or the board game Diplomacy [5].

In this paper we introduce a new family of Branch and Bound algorithms, namely NB^3, that use negotiation as the key element in the exploration of the joint space of solutions for a number of autonomous agents. Section 1 briefs on Branch and Bound algorithms. Section 2 explains the concept of the algorithm. In section 3 we describe a new problem that we defined specifically for testing negotiation algorithms like NB^3, and in section 4 we present some initial experiments we have performed to test the algorithm and we give their results. Finally, in Section 5 we conclude.

1. Branch & Bound algorithms in a nutshell

Branch&Bound (BB) is a general algorithm to find optimal solutions in discrete domains. Here we outline the basics of the algorithm and introduce some notation, for an in-depth description we refer to [6,7].

The objective of a BB algorithm is to find a solution x to a problem that minimizes (or maximizes) a given function $f(x)$. The algorithm incrementally generates a tree where nodes represent sets of solutions S. Children nodes represent subsets of the father $(S_1, \ldots, S_i, \ldots, S_n)$ forming (ideally) a partition. A BB algorithm consists of three basic operations. The first one is to split the set of solutions represented in a node into a number of subsets that become the children of the node. This operation is called *Branching*. It is clear that $min_{x \in S}\{f(x)\} = min_{S=\cup_i S_i}\{min_{y \in S_i} f(y)\}$. The second operation is the establishment of bounds, lower and upper, for the value of $f(x)$ on the elements of S_i. This step is called *Bounding*. These bounds indicate how close we are to the optimal solution.

Finally, the key idea in any BB algorithm that looks for a minimum of $f(x)$ is that if the lower bound of a node is higher than the upper bound of another node, then the former node can be ignored as it will not contain the optimal solution. This third step is the *pruning* of the tree. This recursive procedure stops when the set S contains a single element or when the lower and upper bounds get equal, i.e. all contained solutions are equally good (or bad).

An advantage of BB is that, as it progresses, the global bounds get reduced and when their interval is reduced to a reasonable size we can stop the algorithm and pick up one element randomly from the set represented by the node.

2. NB^3 basic concept

Branch and bound has mostly been used as a centralized algorithm. Distributed versions do also exist that try and exploit concurrency in the exploration of the tree [8]. However, not much work has been done on the application of BB algorithms in search problems where the splitting is based on variables that are controlled by different agents, as in the asynchronous backtracking method used in Distributed Constraint Satisfaction [2], and where there is no single function $f(x)$ to optimize but a *set* of functions, one per agent, that are not centrally known. This paper proposes an algorithm that is run by every agent in a multiagent system and that uses negotiation between agents to split and to prune nodes. Next we give the basic idea of the algorithm (for more details we refer to [9]).

We assume a number of agents $A = \{\alpha, \beta, \ldots, \omega\}$ situated in an environment $\epsilon \in \mathcal{E}$. Each agent[1] $i \in A$ has the capacity of executing a set of feasible actions in a particular environment and given a set of known commitments[2]. This set of feasible actions is denoted by $fea(\mathcal{C}_i, \epsilon) \subset \mathcal{O}$, where $\mathcal{O}$ represents the set of all possible actions by any agent in any environment and $\mathcal{C}_i \subseteq \mathcal{O}$ is the subset of actions that agent i has already committed itself to. For convenience, inaction is considered as a possible action.

At particular time instants, agents decide autonomously what actions to perform in the environment. They are endowed with private goals and thus select those actions that might be more profitable for their goals. We are assuming environments where dependencies between actions are fundamental. In other words, certain actions performed by α will only be successful if they are accompanied by certain actions performed by β. This means that agents cannot decide what to do in isolation. They are assumed to have the capability of persuading one another, via negotiation, in order to co-ordinate their actions. We don't assume any global goal, agents only use their private goals to evaluate plans. We do assume that agents know which other agents are available and their possible actions. A set of actions, that is, a joint plan, $p = O_\alpha \cup O_\beta \cup \cdots \cup O_\omega$, where $O_i \subseteq fea(\mathcal{C}_i, \epsilon)$ for all $i \in A$, executed on an environment ϵ will end up in a new environment ϵ', denoted $p(\epsilon) = \epsilon'$. Agent α will measure how good ϵ' is to its goals with the help of a private function $f_\alpha(\epsilon)$. Instead of a single function to optimize $f(\epsilon)$, as in classical BB, in a multiagent setting we are then dealing with a set of functions $\{f_\alpha(\epsilon), f_\beta(\epsilon), \ldots, f_\omega(\epsilon)\}$ each one being locally optimized by a copy of the NB^3 algorithm.

During the process agents make commitments to perform certain actions by accepting proposals and reject actions by rejecting proposals. Agents have a partial view of the commitments made as conversations may be private.

We next explain the different components of NB^3 from the perspective of agent α.

2.1. Search tree

In a multiagent setting, each agent that runs the algorithm builds its own search tree. The root node of the NB^3 search tree consists of all the possible solutions to the problem. Each agent may choose to perform a subset of its feasible actions $fea(\mathcal{C}_i, \epsilon)$. The set of all feasible actions is denoted $Fea = \bigcup_{i \in A} fea(\mathcal{C}_i, \epsilon)$, and the set of plans[3] represented by a node n, is denoted as $plans(n)$ (which is a set of subsets of Fea). As already mentioned, this is in most practical applications an intractably large set for exhaustive exploration. The children of a node form a partition of the solutions of the father node. We label the link between a father and a son with the name of an action contained in all the solutions represented by the child. A path between the root and a leaf of the tree is then a (joint) (partial) plan (i.e. those actions labeling links in the path) that guarantees, at least, the worst solution in the leaf node. Given the path from node n back to the root node, we denote by $n.path$ the set of actions in the path from all agents in A, that is $n.path \subseteq \bigcup_{i \in A} fea(\mathcal{C}_i, \epsilon)$.

[1] We use Greek letters as the names of specific agents, while we use Latin letters to refer to any undetermined agent.

[2] A commitment is the declaration of an intention to act.

[3] To keep things simple we assume here that the order in which actions are taken is irrelevant for the outcome of the state of the world. Therefore, we see a plan as a set of actions, rather than a sequence of actions. So a set of plans is a set of a set of actions.

We also assume that α is situated in an environment that has strict time limits for a decision to be made. If the tree has been explored completely and an optimal solution has been found before the deadline then the decision of what α has to do is easy: the actions corresponding to α in the path to the optimal leaf node. Otherwise, the set of actions in the path from the root to the node with the best bounds plus the known commitments is a possibly good choice for action, even if only partial, and is what NB^3 considers as the best plan. In that respect, NB^3 is an anytime algorithm that always has the so far *best* plan of action ready.

2.2. Splitting

We are assuming a negotiation environment in which commitments among the participants have to be made along the search process. This is so because an agent cannot wait until it finds the optimal plan before negotiating with other agents, as then it would perhaps be too late to get any commitment from them: they might have already signed commitments for incompatible actions. Therefore, a trade-off exists between optimality and commitment availability. The more we forward explore from a potential commitment of others the better, but then the less probable it is to get it. How to solve this trade-off is key in the application of NB^3 to a particular problem.

Another key element of NB^3 is the decision on which node and according to which actions α should split. The algorithm generates a tree according to a best-first search, in which the best node to expand is determined by a heuristic h. This heuristic is a fundamental parameter of NB^3. It must rank the splits to make at each node according to the path to the node $n.path$ and both the goals and the trust[4] attitude of α.

2.2.1. Offers

When a node is found for which the set of actions in its path to the root is considered as good enough, the agent will (i) issue as many offers as needed to get the commitments from others that are required to execute this plan, and will (ii) withdraw any standing offers that are incompatible with the offers just made.

In particular, when the heuristic h used by α chooses to split a node according to β's actions and one of the children is selected to be proposed, NB^3 issues an offer to β to get its commitment on the action labeling the arc from that child to the father. If there is a standing offer that is incompatible with that action, NB^3 withdraws it.

While waiting for the acceptance of issued offers, NB^3 keeps on expanding the tree. h should prioritize those actions of α that might be interesting to the agents with open negotiation threads in preparation for a counter-offer.

When α receives an offer from β, h should prioritize those actions contained in the offer that are compatible with those in the path to the current best node. In this way, agents help each other in focusing the search on the space of potential deals.

2.2.2. Withdraws

When α receives a withdraw from agent β it prunes all nodes that require those actions of β in the proposal just withdrawn. Also, any withdraw is an indication that the probability

[4] Don't forget that commitments may not be respected.

of reaching a deal with β is lower than before and consequently the actions by β should have less priority in future selections to be made by h.

When α withdraws a previously sent offer to β the probability of reaching agreements with β decreases as β might be unhappy with the decision[5] and then the selection of actions that were potentially favorable to β and that h was going to select early in the splitting process should be delayed.

2.3. Bounding

During the search, the algorithm calculates the following values:

- A *global upper bound*: gub_α. The value of the best environment the agent could achieve from the current world state, without co-operation from any other agent.
- For each node n an *intermediate value*: $e_\alpha(n)$. A value that estimates the value α will obtain if the partial plan in the path $p = n.path$ and the known reached agreements $\mathcal{C}_\alpha$ are executed and no extra actions are done. That is: $e_\alpha(n) = f_\alpha((p \cup \mathcal{C}_\alpha)(\epsilon))$.
- For each node n a *lower bound*: $lb_\alpha(n)$. An estimate of the value of the best environment reachable from the joint plans in n.
- For each other agent an estimation for its global upper bound: $\{gub^\alpha_\beta,\ gub^\alpha_\gamma, ...\}$.
- For each other agent and each node n an estimation for its intermediate value: $\{e^\alpha_\beta(n),\ e^\alpha_\gamma(n), ...\}$
- For each other agent and each node n an estimation for its lower bound: $\{lb^\alpha_\beta(n),\ lb^\alpha_\gamma(n), ...\}$

In our notation we use the convention that estimations of quantities that belong to other agents have two indices. The superscript index refers to the agent making the estimation of the quantity, while the subscript index refers to the agent this quantity belongs to. So for example gub^α_β is defined as the estimation that α makes about β's global upper bound. Since the algorithm in this paper is entirely described from the point of view of agent α we will only encounter estimated quantities for which the superscript index is the letter α.

The global upper bound can be considered as the 'acceptance level': an agent will never accept any deal if it gives him a higher cost than the global upper bound. The world 'global' refers to the fact that this value is not assigned to any specific node in the tree, but rather to the entire tree itself; it is a property of the current world state.

The intermediate value of a node is the value that the agent would get if the actions in the path from this node to the root node are executed. So if $e_\alpha(n) > gub_\alpha$ the plan corresponding to node n is not profitable for α. Therefore we say a node n is *rational* for agent α iff $e_\alpha(n) < gub_\alpha$.

Note that for general BB algorithms in which one tries to minimize, usually each node carries its own local upper bound and the global upper bound is then defined as the minimum of all the local upper bounds. In the case of NB^3 however, we don't have such local upper bounds. The reason for this is that, if we would have local upper bounds and we look at the node with the lowest local upper bound, we cannot be sure that we can actually reach this node, because other agents might not co-operate with the plan

[5] humans may be involved in the MAS.

corresponding to this node. Therefore, we are never guaranteed to be able to reach the node with the lowest local upper bounds, which makes the definition of the gub as a minimum of local upper bounds useless.

When the agent needs to choose a plan to propose, the node with the lowest value of $e_\alpha(n)$ represents the plan with the lowest cost for α so this is the plan that α would prefer to execute. However, in general α needs the co-operation of some other agents to execute it, which means that the plan should also be profitable for them. This means that, as in any other form of negotiation, α should make a trade-off between minimizing its own costs, and the costs of its negotiation partners. For this reason, agent α needs to estimate the global upper bounds, intermediate values and lower bounds of the other agents.

2.4. Pruning

The lower bound is used for pruning: it defines the lowest cost an agent could possibly achieve in any descendant of the node. If $lb_i(n) > gub_i$ for any agent i involved in the actions in $n.path$, not only is this deal unprofitable for agent i, but also any deal that could be found by further exploring the children of node n will be unprofitable for i, so in that case agent i would never agree with any deal descending from node n and therefore this node can be pruned.

Furthermore, when an offer that α has issued is accepted by agent β it means that agent β is making a commitment to perform certain actions from the set of available actions $O \subseteq fea(\mathcal{C}_\beta, \epsilon)$. All actions that are incompatible with those in O are unfeasible, therefore we can prune all nodes that have any of the incompatible actions in their paths to the root. When an offer made by α to β is rejected the nodes that assumed the actions of β in the offer are also pruned.

3. The Negotiating Salesmen Problem

In this section we describe a new variant of the Traveling Salesman Problem (TSP), which we have defined in order to test the NB^3 algorithm. We call this problem the *Negotiating Salesmen Problem* (NSP). It resembles the multiple Traveling Salesmen Problem (mTSP) as described in [10], but with the main difference that each agent in the NSP is only interested in minimizing its individual path.

The idea is that several agents (the salesmen) need to visit a set of cities. The salesmen all start in the same city, and all other cities should be visited by at least one agent. Initially, each city is assigned to one salesman that has to visit it. However, the salesmen are allowed to exchange some of their cities amongst each other, so that the agents might be able to decrease the distance they have to cover. For example: if a city v is assigned to agent α, but α prefers to visit another city v', which is assigned to agent β, then α will propose to β to exchange v for v'. If β however also prefers to have v' over v he will not accept this deal. And if no other agent wants to accept v either, then α is obliged to travel along city v. However, we impose the restriction that not all cities are allowed to be exchanged. The cities that can be exchanged are referred to as the *interchangeable cities*, while the cities that cannot be exchanged are called the *fixed cities*.

In the following, all sets we mention are finite.

Definition 1 *An instance of the NSP is a tuple $\langle G, v_0, A, F, I, \epsilon_0, t_{dead}\rangle$, which consists of: a weighted graph G, a special vertex v_0 of the graph, a set of agents A, a set of fixed cities F, a set of interchangeable cities I, an initial distribution of cities ϵ_0 and a deadline t_{dead}. These components are further explained below.*

G is a complete, weighted, undirected graph: $G = \langle V, w\rangle$ with V the set of vertices (the *cities*) and w the weight-function that assigns a cost to each edge: $w : V \times V \rightarrow \mathbb{N}$ such that it satisfies the triangle inequality:

$$\forall a, b, c \in V : \quad w(a, c) \leq w(a, b) + w(b, c)$$

One of the vertices is marked as the *home city*: $v_0 \in V$. Each agent has to start and end its trajectory in this city. We use the symbol $\bar{V}$ to denote the set of *destinations*, that is: all cities except the home city: $\bar{V} = V \backslash \{v_0\}$. The set of destinations is partitioned into two disjoint subsets: F and I, so: $\bar{V} = F \cup I$ and $F \cap I = \emptyset$. They are referred to as the set of *fixed cities* and the set of *interchangeable cities* respectively.

The set of agents (the *salesmen*) is denoted by $A = \{\alpha, \beta, ...\}$. Each destination is initially assigned to an agent, by the function $\epsilon_0 : \bar{V} \rightarrow A$. We use the symbol $\bar{V}_i$ to denote the subset of $\bar{V}$ consisting of all cities that are assigned by ϵ_0 to agent i. $\bar{V}_i = \{v \in \bar{V} |\, \epsilon_0(v) = i\}$. $\bar{V}_i$ is referred to as agent i's set of preassigned cities. The definitions above imply that for each agent its set of preassigned cities can be further subdivided into: $\bar{V}_i = F_i \cup I_i$ where F_i is defined as $\bar{V}_i \cap F$ and I_i is defined as $\bar{V}_i \cap I$.

Finally, the instance includes a real number t_{dead} which represents the deadline for the negotiations. Agents are allowed to negotiate over the assignment of cities, until this deadline has passed.

Definition 2 *A solution of an instance of NSP is a tuple $\langle \epsilon_s, T\rangle$ in which ϵ_s is a distribution of cities: $\epsilon_s : \bar{V} \rightarrow A$ such that the restrictions of ϵ_0 and ϵ_s to F are equal: $\forall v \in F : \epsilon_0(v) = \epsilon_s(v)$. $T = (T_\alpha, T_\beta, ...)$ is a tuple of finite sequences of cities, one for each agent, such that for each agent i, T_i contains v_0 and all cities in $\bar{V}'_i$, with $\bar{V}'_i = \{v \in V | \epsilon_s(v) = i\}$.*

This means that in the solution, the cities are distributed between the agents according to ϵ_s, but the fixed cities F are still assigned to their original agents (they cannot be exchanged). So in the solution, the cities are redistributed: $\bar{V} = \bar{V}'_\alpha \cup \bar{V}'_\beta \cup ...$, but the fixed cities are not: $\bar{V}'_i \cap F_i = F_i$. A sequence T_i of the solution represents a cycle in the graph that starts and ends in v_0 and that passes all vertices in $\bar{V}'_i$. For each agent we then have a cost: $c(T_i) \in \mathbb{N}$, which is the length of the cycle. If T_i is given by $T_i = (v_0, v_1, v_2, ...v_k)$, then $c(T_i)$ is defined as:

$$c(T_i) = \sum_{j=1}^{k} w(v_{j-1}, v_j) + w(v_k, v_0) \tag{1}$$

By definition, an agent i prefers a cycle T_i^1 over a cycle T_i^2 if and only if $c(T_i^1) < c(T_i^2)$. We assume all agents are rational and therefore a solution is only feasible if for each agent the cost of the solution is less then the cost it would incur from the original distribution of cities ϵ_0. Note that, because of the fact that the graph is complete and satisfies the

triangle inequality, one can always assume that the shortest path through a given subset of cities goes only through these cities, and does not pass any other city. Therefore, an agent can limit its search to paths that only visit its own cities.

4. Experiments

We have implemented an agent that applies the NB^3 algorithm to the NSP and performed some initial experiments with it, of which we will now present the results. It should be noted however, that at this moment no attempt has been made to implement the algorithm as efficient as possible. The results presented here were obtained with naïve heuristics and inefficient implementation. We therefore expect that the results can be improved a lot in the future.

We have done 3 experiments, each with 5 agents simultaneously running the algorithm. For each experiment we have created 5 instances of the NSP and for each such instance we have repeated the algorithm 3 times. So each experiment consisted of 15 runs with 5 agents simultaneously running the algorithm.

The three experiments differ in the number of interchangeable cities that were assigned to the agents. In the three experiments each agent had 5, 7 and 9 interchangeable cities assigned to it, respectively. Furthermore, in each experiment each agent had 1 fixed city, and the deadline for the negotiations was set at 60 seconds.

4.1. Creation of Problem Instances

For our experiments we created a few problem instances with obvious optimal solutions. The graphs of these problem instances are Euclidean graphs. That is: each city corresponds to a pair of 2-dimensional coordinates, and the distance between two cities is simply the 2-dimensional Euclidean distance.

Each graph is created as follows: we first create l random cities (with l the number of agents: $l = |A|$), far away from each other. Each of these cities is assigned to one of the agents as a fixed city (each agent gets exactly one fixed city). For each such fixed city we then generate m cities nearby. In this way we have created l clusters of each $m + 1$ cities. At first, all the cities of one cluster are assigned to the same agent. We refer to this assignment as the *'optimal assignment'* (it is optimal in the sense that each agent owns a set of cities which are very close to each other, so that the distance each agent has to cover is minimal). Then, for each agent we randomly choose one of its cities and interchange it with a random city from any another cluster.

After these changes each agent owns at least one city from another cluster, and on average each agent owns two cities from another cluster (because we make l swaps, and each swap involves 2 cities involved, in total $2l$ cities change owner). We refer to this new assignment as the *'initial assignment'*, because this is the assignment of the cities at the start of the algorithm. So the goal of the algorithm is to retrieve the optimal assignment from the initial assignment.

4.2. Evaluation of Results

While running the algorithm, every time when a deal between some of the agents is confirmed, we store the time of confirmation, and for each agent, the set of cities it owns

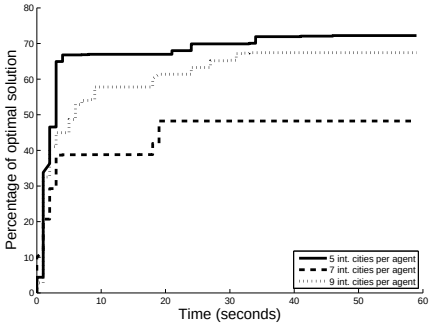


Figure 1. Results

after the deal. Also we store the initial assignment and the optimal assignment (so if k deals have been made during the execution, $l \cdot (k+2)$ sets of cities are stored).

After the algorithm has finished, for each of these sets of cities, we find the shortest path through these cities, by feeding them into the *Concorde TSP Solver* [11]. The length of the shortest path through the set of cities owned by agent i at time t is then denoted as $C_i(t)$. The length of the shortest path through the set of cities that are assigned to agent i in the optimal assignment, is denoted by C_i^*. With this notation we then define our performance measure as the following quantity:

$$Q(t) = \frac{100}{l} \sum_{i \in A} \frac{C_i(0) - C_i(t)}{C_i(0) - C_i^*} \tag{2}$$

Notice that at $t = 0$ its value is 0 by definition, and that when the agents have succeeded in retrieving the optimal assignment its value is 100 (we have included the factor of 100 in the definition so we can interpret it as a percentage).

The results of the three experiments are shown in Figure 1. What we see is that at least in two of the three experiments the algorithm has been able to find most of the deals that were necessary to retrieve the optimal assignment of cities. The fact that not all of them have been found is probably due to the fact that the algorithm uses relatively bad heuristics for estimating the length of the path through a set of cities. We expect that this can be improved.

One would expect that the results get worse as the number of cities increases, because the search space gets larger. We see however that the experiment with 7 interchangeable cities has worse results than the experiments with 5 and with 9 agents. We think that this is caused by the fact that we have used too few problem instances, so that some of the instances with 7 interchangeable cities per agent are, by coincidence, a bit more complex than the other ones. Therefore, we expect that this effect disappears when we use more instances.

5. Conclusions and Further Work

From the experiments we can conclude that the algorithm clearly works, although it is not yet optimal. However, since the results presented here were obtained using a very early

version of the algorithm, there is much room for improvement. Especially we expect that we can drastically decrease the time necessary to reach agreements, by writing the code in a more efficient way.

The instances of the NSP that we used here were very simple instances, because they have obvious solutions. The cities are clearly divided in equally sized clusters, so each agent should simply collect the cities in the cluster where it has its fixed city. It would be interesting to see how the algorithm behaves with more complex problem instances.

More specifically, we should apply the algorithm to instances that involve a certain amount of competition between the agents. In the current instances, the optimal solution is highly satisfying for each agent individually, so that the problem can in fact be seen as a Distributed Constraint Optimization Problem, a kind of problem for which many good algorithms have already been invented. The point of our algorithm is however, to solve problems in which the individual wishes of the agents are not compatible. For this we first need to improve the heuristics.

Furthermore, we should compare our algorithm with other negotiation algorithms. The problem is however, that almost no algorithms have been developed so far that are capable of negotiation in large search spaces. So instead we could compare our algorithm with simplified versions of the same algorithm.

Finally, we would like to test NB^3 in other environments than NSP. For example in the game of Diplomacy (which has a huge search space) and in the problem of negotiating over time tables (in which 'preference' is a more abstract, non-numerical quantity).

Acknowledgements

Research supported by the Agreement Technologies CONSOLIDER project under contract CSD2007-0022 and INGENIO 2010 and by the COST Action, IC0801.

References

[1] Nick Jennings, Peyman Faratin, Alessandro Lomuscio, Simon Parsons, Carles Sierra, and Mike Wooldridge. Automated negotiation: Prospects, methods and challenges. *International Journal of Group Decision and Negotiation*, 10(2):199–215, 2001.

[2] Makoto Yokoo, Edmund H. Durfee, Toru Ishida, and Kazuhiro Kuwabara. The distributed constraint satisfaction problem: Formalization and algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 10:673–685, 1998.

[3] RJ Willemen. *School timetable construction : algorithms and complexity*. T.U. Eindhoven, 2002.

[4] MJC^2. http://www.mjc2.com/transport_logistics_management.htm.

[5] Angela Fabregues and Carles Sierra. Dipgame: a challenging negotiation testbed. *Engineering Applications of Artificial Intelligence*, 2011.

[6] E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations Research*, 14(4):699–719, 1966.

[7] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[8] B. Gendron and T. G. Crainic. Parallel branch-and-bound algorithms: Survey and synthesis. *Operations Research*, (42):1042–1066, 1994.

[9] Carles Sierra. Nb3: Negotiation-based branch and bound. Technical report, IIIA-CSIC, Bellaterra (Barcelona), Spain, 2011.

[10] Tolga Bektas. The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34(3):209–219, June 2006.

[11] D. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook. http://www.tsp.gatech.edu/concorde.

Towards plant monitoring through Next Best View

Sergi FOIX [1], Guillem ALENYÀ and Carme TORRAS
Institut de Robòtica i Informàtica Industrial, CSIC-UPC

Abstract. Monitoring plants using leaf feature detection is a challenging perception task because different leaves, even from the same plant, may have very different shapes, sizes and deformations. In addition, leaves may be occluded by other leaves making it hard to determine some of their characteristics. In this paper we use a Time-of-Flight (ToF) camera mounted on a robot arm to acquire the depth information needed for plant leaf detection. Under a Next Best View (NBV) paradigm, we propose a criterion to compute a new camera position that offers a better view of a target leaf. The proposed criterion exploits some typical errors of the ToF camera, which are common to other 3D sensing devices as well. This approach is also useful when more than one leaf is segmented as the same region, since moving the camera following the same NBV criterion helps to disambiguate this situation.

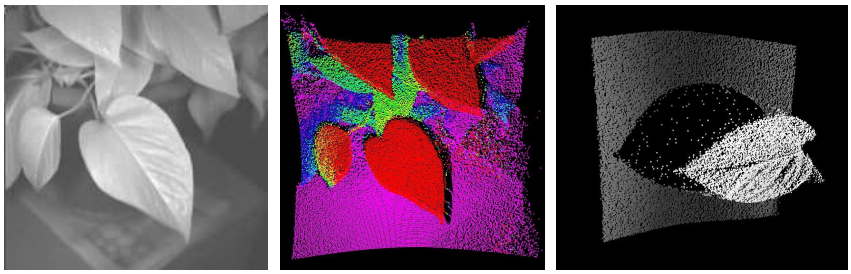
Keywords. Next Best View, ToF cameras, depth images, plant segmentation, leaves disambiguation

Introduction

Food industry is very important for society, and large areas of the world are currently cultivated, as open plantations or as greenhouses. The automation in such areas has been traditionally intensive, generally at large scale and relying on human assistance. Recently, more attention is given to standalone processes taking increasingly into account plants as individuals [1]. In the context of the GARNICS project, we aim at the monitorization of large plantations to help determine the best treatments (watering, nutrients, sunlight) to optimize pre-defined aspects (growth, seedling, flowers) and eventually guiding robots to interact with plants in order to obtain samples from leaves to be analysed or even to perform some prunning.

Monitoring and taking actions over plants are two very difficult tasks. The reason why these tasks are so difficult is because plants are complex and dynamic systems. Two plants are not equal. They are composed of multiple elements such as flowers, leaves, stem and roots. They grow, changing their shape and incorporating new elements. They move and they change their colors depending not only on intrinsic but also extrinsic components. Because of all of these plant behaviours, tasks such as feature detection and action planning over them are very hard problems to solve.

[1]Corresponding Author: Sergi Foix, Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028 Barcelona, Spain; E-mail: sfoix@iri.upc.edu



(a) Intensity image (b) Depth colored 3D point cloud (c) False depth points at edges

Figure 1. Typical images acquired with a ToF camera (200×200 PMD CamCube 3.0). Interesting false depth measures appear at the edges between foreground and background due to the integration of the reflected light of both surfaces in the corresponding pixels.

But leaves are not uniformly arranged in space, albeit they grow, by their nature, in a very structured way. Therefore, monitoring and measuring actual properties of the plant over its leaves requires of specific tecniques in order to place a sensor into the correct pose. The next-best-view algorithm presented in this article focuses its attention to find a point of view that provides a better perception of an occluded leaf or, in a similar way, to disambiguate about the quantity of observed leaves. At the same time and as a consequence, a better estimation of the leaves poses is achieved, a necessary requirement to achieve the placement of a tactile measuring tool over the leaf.

Plants are a hard scenario for segmentation algorithms based on traditional color vision, mainly due to the lack of texture and the uniformity of color. It has recently been demostrated that 3D information is highly valuable in this context [2]. Such information is obtained with a depth sensor, that should provide information independently of the illumination conditions, as they change in greenhouses. Acquisition time is also important, as a lot of plants should be monitorized. Finally, the sensor has to be lightweight, as we want to mount it in the end effector of a robotized arm. Time-of-Flight cameras are lightweight 3D cameras that provide directly depth images without pre-processes, with infrared autoillumination units, that deliver 30 frames per second.

The article is structured as follows: in Sec. 1 the 3D image acquisition through ToF cameras is introduced. Section 2 explains the proposed Next-Best-View algorithm and how it takes advantage of erroneously captured data. This algorithm is validated in Sec. 3, including also some considerations about the camera and scene configuration. Finally, in Sec. 4 conclusions and future work are presented.

1. 3D image acquisition

Depth measurements are carried out by a relatively new type of sensor named Time-of-Flight (ToF) camera. This type of sensor has the main characteristic of providing registered depth and intensity images of a scene at a high frame-rate (see Fig. 1(a) and 1(b)). ToF cameras use the well-known time-of-flight principle to compute depth. The camera emits modulated infra-red light in order to measure the travelling time between the known emitted waves and the ones reflected back over the objects in the scene. Compared to other similar technologies, such as the new Kinect, and taking into account the

context of the GARNICS project, ToF cameras provide some interesting features that make them more suitable for short range applications. It has auto-illumination, making it independent from external light sources, and its minimal depth measuring range can get as close as 15 cm. [2]

But ToF-cameras have two main drawbacks: low resolution (200×200 pixels for a PMD CamCube 3.0 camera) and noisy depth measurements due to systematic and non-systematic errors. On the one hand, low resolution can be a big problem for large environment applications, but it has not such a negative impact when the camera is used at 20 cm range as it is our case[3]. On the other hand, noisy depth measurements due to non-systematic errors get amplified by working in such a short range. Mainly the ones due to multiple light reception and light scattering. Systematic errors get highly reduced by calibration procedures [3]. For a more detailed and wide classification and explanation of the different error sources, advantages and limitations of ToF cameras, please refer to [4].

There is one type of multiple light reception error that deserves special attention in this article. This is the jump-edge error (Fig. 1(c)). This type of error appears due to the mix of measurements over the pixels that contain the edges between foreground objects and their background, refer to Sec. 2.2 for a more detailed explanation. Our approach takes advantage of detecting this type of error on the scene, and computes a new next-best-view in order to acquire a better estimation of the leaves composition. Jump-edge errors are not unique of ToF cameras but are also present in lidar systems and the new Kinect[4].

2. Improving 3D information through Next Best View

Next-best-view (NBV) is one of the most challenging problems in vision sensor planning. Its application covers tasks such as autonomous 3D object modelling, object recognition, visual tracking or, as in our case, monitoring complex systems. Initial investigations in the field of NBV were presented in [5], giving two algorithms to determine best next views that established the basis for further research: the planetarium algorithm (slower due to consider possible occlusions), and the normal algorithm (much faster, but weaker with occlusions). Subsequent research studied the use of camera triangulation systems, and in [6] the use of a range scanner was suggested. The authors concluded that using depth information from range data into the NBV problem was a tool for cost-effective and accurate acquisition of 3D data. More recently, in [7], a method is proposed for automatically acquiring 3D models of unknown objects by moving the sensor around the target object. Sensor motion is determined by the analysis of the curvature's trend at the surface edges.

The level of difficulty in NBV does not depend only on the task but also on some common aspects such as: whether a prior model of the object is known or not, whether a very precise range sensor is used or not, and whether the viewpoint working space is highly constrained or not. In this work we assume that plants are composed of nearly

[2] Measures extracted with a PMD CamCube 3.0 camera after changing its modulation frequency to 21MHz and decreasing its integration time to 0.2 ms.

[3] 20 cm ensures a good compromise between planar model fitting and signal-to-noise ratio.

[4] Due to its internal filtering, Kinect does not deliver these data.

planar leaves so we rely on planar models, a noisy 3D range sensor is used and the viewpoint working space is constrained by the manipulator robot working space and by the pre-defined maximum distance between the camera and the surface of the plant.

Although the following sections give a more comprehensive explanation of each of the steps in the view sensor planning, here is a brief summary. Initially, the camera is placed at approximately 15-20 cm away from the plant's region of interest. Secondly, leaves are segmented by means of planar approximation. Thirdly, jump-edge points are detected. And finally, by combining the data from the previous two steps, the NBV is computed.

2.1. Leaf segmentation - Fitting planar models to leaves

Each plant has its own specific type of leaves and their shapes and sizes can vary in a wide range. Although more accurate leaf 3D models can be defined and consequently improve the detection of leaves and the estimation of their poses, in our approach a simple planar model has been used. Fitting accurate 3D object models to crowded scenes is a very time consuming task, and it gets worse when the data provided are noisy as it happens in the case of ToF cameras. Consequently, and when plants have nearly planar leaves, simple plane models can be approximated and therefore increase the speed of 3D data processing.

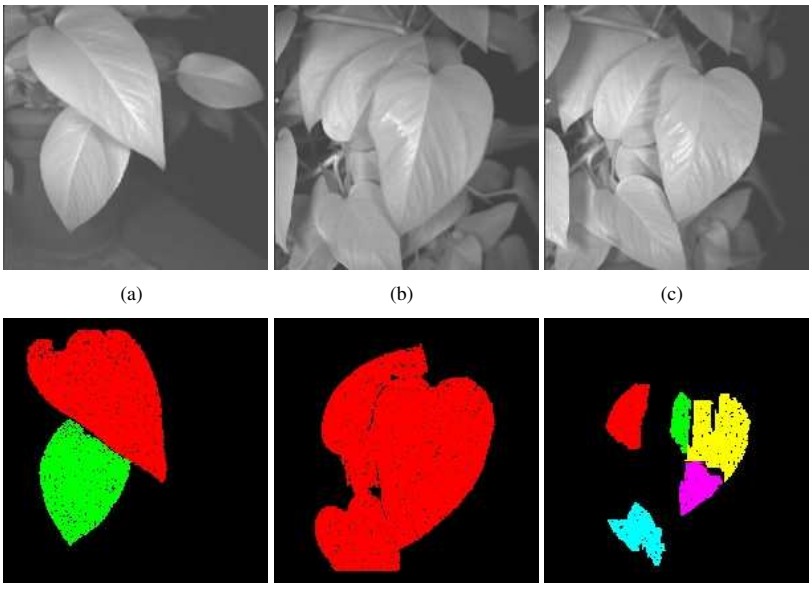
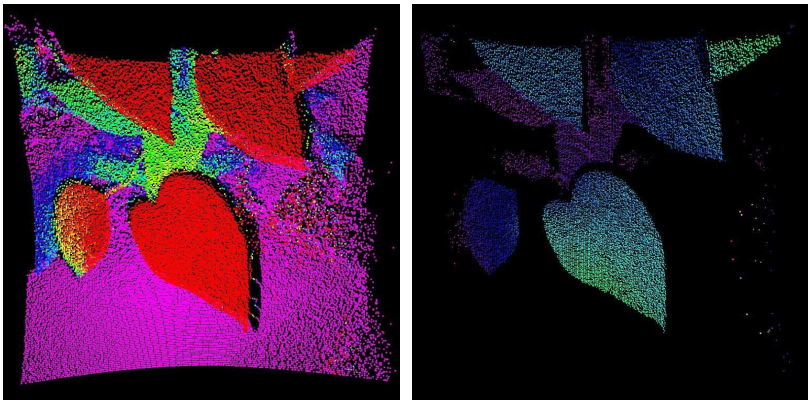


Figure 2. Planar leaf segmentation is highly parameter-dependent. The first row shows some intensity images, while the second row shows their corresponding planar segments defined by colors. Images (a,d) and (b,e) share the same parameterization. It is possible to see how we obtain different segmentation results for the same parameterization. Images (c,f) are the same scene as (b,e) but with different parameterizations. Here it is possible to see how a bad leaf segmentation is produced due to the non-planar shape of one of the leaves.

But there are always some drawbacks. Planar leaf segmentation is a highly parameter-dependent algorithm. Depending on the shape of the sensored surfaces that need to be modelled and the quality and density of the acquired 3D data, necessary pre-processes for plane estimation, such as point-normal calculation and point-neighbourhood computation, can be very tricky to tune. In the case of plants with planar leaves, where data is captured with a ToF camera, these tunning parameters have to allow dealing with the highly noisy readings from the sensor and try not to subdivide a single leave in multiple planes. An example of a bad parameterization can be observed in Fig. 2(f). It is preferable fusing two leaves as if they were a single one than subdividing a single leaf in sub-elements. This is because, as it has been said previously and will be demonstrated by experiments in Sec. 3, ambiguity can be resolved by acquiring a new best view.

2.2. Jump-edge filter

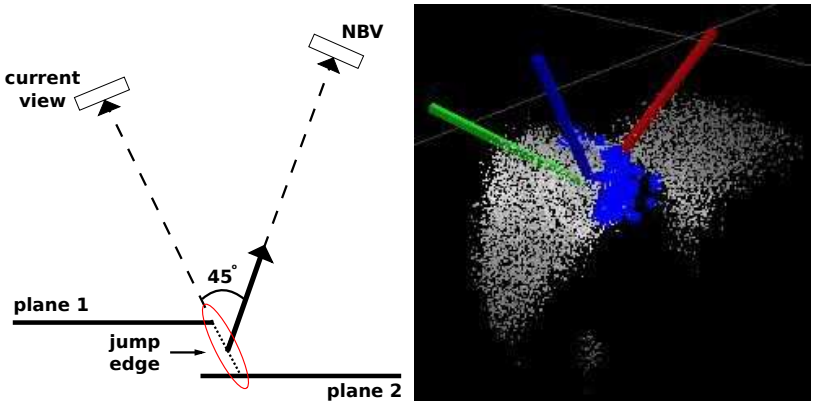
Figure 3 shows the appearance of a curtain of flying points around the edges between foreground objects and their background. These points are commonly known as jump-edge points and are generally removed by comparing the angle of incidence of neighboring pixels [8,9,10]. They are false measurements and consequently they are always removed from the data sets, even the new Kinect sensor filters internally these misreadings. But in our case the appearance of these false measurements are indicative of possible model misinterpretation or object occlusion. Therefore, their detection and 3D localization in the scene provide the required information for computing the next-best-view that will try to disambiguate or improve occluded leaf visibility and pose estimation. In our algorithm, a number of at least 20 jump-edge points have to be detected in order to consider them a region of interest. This threshold has been set empirically to prevent considering non-systematic noise as jump-edge points.



(a) Raw 3D points colored by depth (b) Filtered 3D points colored by depth

Figure 3. Comparison between raw and filtered 3D point clouds. Image (a) clearly shows how raw measurements incorporate undesired data into the 3D point cloud. A curtain of points can be identified on the edges between the foreground (leaves) and the background. Image (b) shows the 3D point cloud after the jump-edge and bounding-box filters have been applied.

2.3. Next position computation



(a) 2D NBV schematic representation (b) Jump-edge points (Blue)

Figure 4. False depth measurements (jump-edge points) detection helps to compute the NBV to uncover occluded leaves and to disambiguate the number of observed leaves. Figure (a) shows the 2D schematic representation of the algorithm. Figure (b) shows, in blue, the 3D jump-edge points.

Figure 4(a) shows a schematic representation of the computation of the NBV for the tasks of uncovering occluded leaves and leaves disambiguation. The main characteristic of our NBV method is that it takes advantage of erroneous depth readings (Fig. 4(b)) for computing a better view in a geometrical way.

Once the overall estimated planes and jump-edge points have been obtain, the computation of the NBV is reduced to a geometrical problem. As introduced in previous sections, the NBV is only calculated if there are jump-edge pixels adjacent to two planes or if these are contained inside a unique plane. For any of both conditions the algorithm behaves in the same manner. First, the median point of the jump-edge points that fulfill the condition is calculated and normalized as a unitary vector. This vector represents the current view camera direction. Second, we calculate the cross product between the estimated plane normal [5] and the previous normalized vector. The resulting orthonormal vector is the one that will act as a rotation axis to attain the NBV (on the schematic representation, this vector would come out from the figure). Finally, using the median jump edge point as a center and the previous rotation axis, a rotation of 45 degrees is applied to the current view. Although 45 degrees have proven to be an adequate quantity in our experiments, it is advisable to use smaller angles, e.g. 10 degrees, since incremental NBV is more adaptative. It has to be noticed that the current method guarantees a gain of information over the scene on superficial leaves but not on the ones deep inside the plant, since their probability of being occluded by unobserved leaves is very high.

[5] In the task of resolving leaf occlusion the normal vector is the one of the occluding plane (closer to the camera).

Figure 5. WAM arm used in the experiments holding the Time-of-Flight camera observing a plant.

3. Experiments

Figure 5 shows the experimental setup of our simulated monitoring plant process. It includes a PMD CamCube 3.0 ToF camera mounted as an end-effector of a 7-DoF Barrett WAM arm. This configuration permits moving the camera to different viewpoints and also monitoring several plants located in the typical matrix-like plant containers.

As it has been previously stated, our proposed NBV algorithm has been designed in order to deal with two specific tasks, resolution of leaves occlusions and disambiguation between leaves. Figures 6 and 7 show two scenes where both tasks have been performed respectively. Each figure is divided in two sets of images, the images at the top row show the state of the scene before applying the NBV algorithm while the images at the bottom row show its state afterwards. By observing the intensity images of the plant it is easy to imagine how common these two types of scenes are obtained in a plant monitoring process and, consequently, how important it is to be able to deal efficiently with occlusions and ambiguities.

Figure 6(a) shows the intensity image of a scene where the occlusion of a leaf is clearly identified. By executing the jump-edge filter over the 3D data, the countours of each leaf are extracted (Fig. 6(b)). At the same time, the plane segmentation process provides the estimation of the different planes (Fig. 6(c)). Figure 6(d) shows, in a 3D rotated view, the extracted jump-edge points that fall just in the frontier between both leaves. These points are the ones that allow us to compute the NBV whose result is displayed at the bottom row of Fig. 6. By comparing the image pairs Fig. 6(a, e) and Fig. 6(c, g), it can be seen by moving the camera to the NBV the overall perception of the occluded leaf surface is significally improved.

Figure 7 shows the ambiguity scene where two leaves have been misinterpreted as only one. In order to evaluate whether there is an ambiguity, the existence of jump-edge points inside the segmented plane is verified. Fig. 7(b) shows how part of the jump-edge points, white contours, are found inside the area of the wrongly assumed leaf (Fig. 7(c)). Following the same NBV approach as before, a new camera pose is computed leading to the resulting images at the bottom row. After the robot's movement, the previously estimated dark red plane has now been correctly divided into two different planes, as it

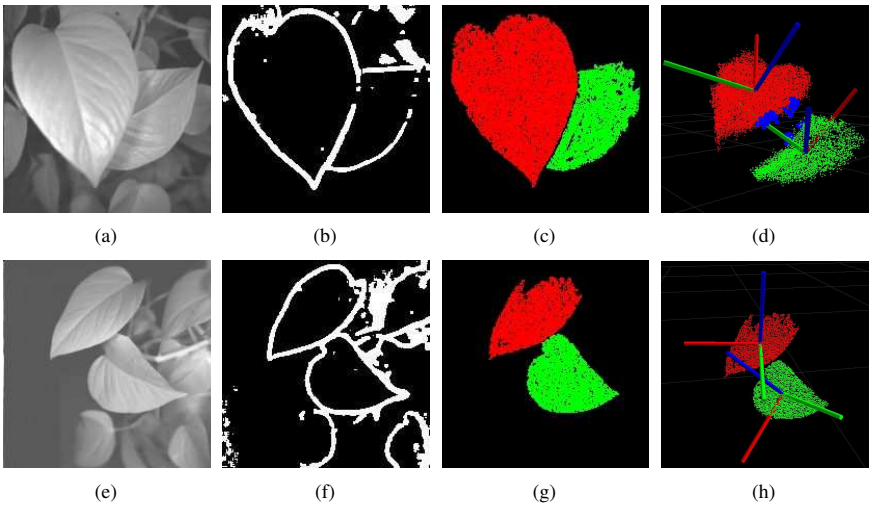


Figure 6. Scene containing a detected leaf occlusion. Top row shows the scene before applying the NBV algorithm, images (a-d). Bottom row shows the scene observed from the new viewpoint, images (e-h). After applying the NBV algorithm the occluded leaf is clearly discovered.

was expected (Fig. 7(g)). Figures 7(d, h) show the final 3D point cloud of the leaves as if they were viewed from the same camera pose, before and after the NBV. It can be clearly seen how not only the disambiguation has been achieved but also how part of one of the leaves that was occluded is now uncovered.

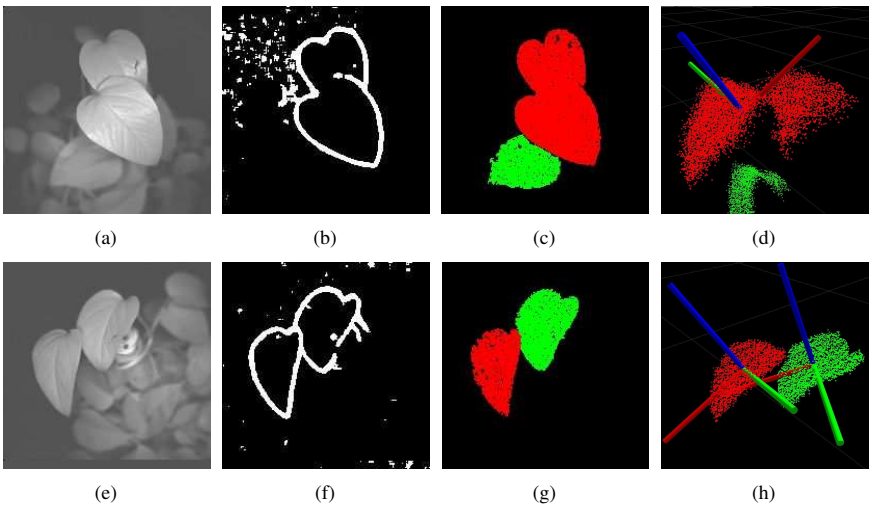


Figure 7. Scene containing a possible mixture of leaves. Top row shows the scene before applying the NBV algorithm, images (a-d). Bottom row shows the scene after it, images (e-h). After applying the NBV algorithm the ambiguity is clarified and two leaves are detected instead of one.

4. Conclusions and future work

This paper proposed a novel method to efficiently estimate a NBV for improving plant monitoring. The method takes advantage of jump-edge flying points, typical erroneous data from a ToF camera, for finding a suitable solution to two common monitoring tasks, getting a better view of an occluded target leaf and resolving ambiguity in the number of leaves. The method can be executed in real-time since it does not use any cost function minimization approach or any complex leaf model fitting but a geometrical approach and a simple planar leaf model.

It has to be noticed that, depending on the configuration of leaves, it may not be possible to completely avoid occlusions or ambiguities by moving the camera. Next research steps will focus on using robot manipulation to help monitoring tasks.

Acknowledgements

This research is partially funded by the EU GARNICS project FP7-247947, by the Spanish Ministry of Science and Innovation under projects DPI2008-06022 and MIPRCV Consolider Ingenio CSD2007-00018, and the Catalan Research Commission through the Robotics Group. S. Foix is supported by a PhD fellowship from CSIC's JAE program.

References

[1] R.D. King, J. Rowland, S.G. Oliver, M. Young, W. Aubrey, E. Byrne, M. Liakata, M. Markham, P. Pir, L.N. Soldatova, A. Sparkes, K.E. Whelan, and A. Clare. The automation of science. *Science*, 5923(324):85–89, 2009.

[2] G. Alenyà, B. Dellen, and C. Torras. 3d modelling of leaves from color and tof data for robotized plant measuring. In *Proc. IEEE Int. Conf. Robot. Automat.*, pages 3408–3414, Shanghai, May 2011.

[3] S. Fuchs and G. Hirzinger. Extrinsic and depth calibration of ToF-cameras. In *Proc. 22nd IEEE Conf. Comput. Vision Pattern Recog.*, volume 1-12, pages 3777–3782, Anchorage, June 2008.

[4] S. Foix, G. Alenyà, and C. Torras. Lock-in Time-of-Flight (ToF) cameras: a survey. *IEEE Sensors J.*, 2011. to appear.

[5] C. I. Connolly. The determination of next best views. In *Proc. IEEE Int. Conf. Robot. Automat.*, volume 2, pages 432–435, St. Louis, Mar. 1985.

[6] Y. Zhien, W. Ke, and Y. Rong-Guang. Next best view of range sensor. In *IEEE 22nd International Conference on Industrial Electronics, Control, and Instrumentation (IECON)*, volume 1, pages 185–188, Aug. 1996.

[7] S. Kriegel, T. Bodenmüller, M. Suppa, and G. Hirzinger. A surface-based next-best-view approach for automated 3D model completion of unknown objects. In *Proc. IEEE Int. Conf. Robot. Automat.*, Shanghai, May 2011.

[8] S. Fuchs and S. May. Calibration and registration for precise surface reconstruction with time of flight cameras. *Int. J. Int. Syst. Tech. App.*, 5(3-4):274–284, 2008.

[9] T. Kahlmann and H. Ingensand. Calibration and development for increased accuracy of 3D range imaging cameras. *J. Appl. Geodesy*, 2(1):1–11, 2008.

[10] W. Karel, P. Dorninger, and N. Pfeifer. In situ determination of range camera quality parameters by segmentation. In *Proc. 8th Int. Conf. on Opt. 3D Meas. Tech.*, pages 109 – 116, Zurich, July 2007.

A Distributed Norm Compliance Model

Ignasi Gómez-Sebastià, Sergio Alvarez-Napagao and Javier Vázquez-Salceda [a,1]
[a] *Knowledge Engineering and Machine Learning Group (KEMLG), Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya*

Abstract. Norms can be used in the scope of distributed computational systems to provide reliable contexts of interaction between parties where acceptable behaviour is specified in terms of regulations or guidelines. These have been explored in various formalisations, and many are the theoretical frameworks that allow to implement and operationalise them. However, when applying these frameworks to complex, heterogeneous scenarios with multiple agents involved, the performance of norm compliance systems may suffer due to bottlenecks, not only in the number of events received, but also on the number and the complexity of the norms being verified. In this paper we present a formal method to distribute norms through a distributed normative monitoring system, based on production systems for maximum efficiency, and a grounding on Strongly Connected Components.

1. Introduction

With the continuous redefinition and evolution of distributed systems –e.g., Multi-Agent Systems, Grid Computing, Service-Oriented Architectures, Cloud Computing–, there is a growing need for these systems' governance. In [5] governance is defined as the distributed control of complex policies in order to ensure the coherence and the stability of these distributed systems as a whole.

There are several abstractions at different levels of expressivity that try to tackle this problem in real-world scenarios, interpreting events as symbolic facts rather than limiting themselves to pure numerical metrics. One of such abstractions is Normative Systems. Research in Normative Systems focuses on the concepts of norms and normative environments[2] in order to provide normative frameworks to restrict or guide the behaviour of (software) agents. The main idea is that the interactions among a group of such agents are ruled by a set of explicit norms expressed in a computational language representation that agents can interpret. Although some authors only see norms as inflexible restrictions to agent behaviour, others see norms not as a negative, constraining factor but as an aid that guides the agents' choices and reduces the complexity of the environment, making the behaviour of other agents more predictable.

In [2] we proposed a reduction from expressive norms to general production systems to build a norm monitoring mechanism that can be used both by agents to perceive the current normative state of their environment, and for these environments to detect norm violations and enforce sanctions. With such a reduction, an agent can configure,

[1] c/Jordi Girona 1-3, E08034, Barcelona, Spain e-mail: {igomez, salvarez, jvazquez}@lsi.upc.edu
[2] Usually called *institutions*.

at a practical level, the production system at run-time by adding abstract organisational specifications –regulative norms– and sets of counts-as rules –constitutive norms. Therefore, in our approach, the detection of normative states is a passive procedure consisting in monitoring past events and checking them against a set of active norms. This type of reasoning is already covered by the declarative aspect of production systems, so no additional implementation in an imperative language is needed. Using a forward-chaining rule engine, events will automatically trigger the normative state –based on the operational semantics– without requiring a design on how to do it.

An advantage of using general production systems is that the efficiency of the system is bound to the complexity of such systems, which is linear to the number of productions contained in the rules in the worst case and constant in the best case [8]. However, in real-world scenarios this might not even be sufficient, as there may be bottlenecks on both 1) the number of events received, and/or 2) the number of norms –and therefore, the number of rules– loaded in the production system. In this paper we focus on the latter and present a model for distributing the normative context among several monitoring systems at run-time. The objective is to effectively reduce the number of events to be taken into account by each monitor and allow each of them to process a smaller part of the whole normative context. Distributed monitors are linked in order to collectively infer the full normative state of the context, and the model ensures that these links are kept to a minimum amount in order to minimise dependencies.

From a practical perspective, we intend to follow the model of distributed interpretation introduced by Lesser et al. [10]: autonomous local nodes with a separate knowledge base, each one responsible of their own *area-of-interest*, and with a decentralised coordination. If we see normative contexts as interpretation areas of interest, our system will provide a cooperative interpretation of brute events as relevant facts from a governance point of view, and thus will allow for norm compliance with high number of norms and/or events to be handled efficiently by taking advantage of the resources of a distributed system. Coordination issues between nodes, such as conflict resolution, are out of the scope of this paper due to space constraints, but we refer to [10] for general ideas on how they can be tackled. In the current paper, we focus on how to split a normative context into separate smaller normative contexts, that is, into separate areas of interest.

This paper is structured as follows: first, an overview of the formalism used to define the normative monitor is provided. Secondly, this formalism is extended in order to detect information (event) dependencies among the components of the normative context, and among monitoring systems. The paper goes on by introducing an approach for dividing a normative context among several monitors. Later, related work is analysed and compared to the proposal presented in this paper. Finally authors' conclusions are provided and future work is outlined.

2. Normative Model

In this section, we introduce the formalism for monitoring normative systems which we will use in the rest of the paper. For more details on this formalism, please refer to [2].

We assume the use of a predicate based propositional logic language $\mathcal{L}_O$ with predicates and constants taken from an ontology O, and the logical connectives $\{\neg, \vee, \wedge\}$. The set of all possible well-formed formulas of $\mathcal{L}_O$ is denoted as $wff(\mathcal{L}_O)$ and we assume

that each formula from $wff(\mathcal{L}_O)$ is normalised in Disjunctive Normal Form (*DNF*). Formulas in $wff(\mathcal{L}_O)$ can be partially grounded, if they use at least one free variable, or fully grounded if they use no free variables.

We define the *state of the world* s_t as the set of predicates holding at a specific timestamp t, where $s_t \subseteq O$, and we will denote $\mathcal{S}$ as the set of all possible states of the world, where $\mathcal{S} = \mathcal{P}(O)$. We will call *expansion* $F(s)$ of a state of the world s as the minimal subset of $wff(\mathcal{L}_O)$ that uses the predicates in s in combination of the logical connectives $\{\neg, \vee, \wedge\}$. We define a substitution instance $\Theta = \{x_1 \leftarrow t_1, x_2 \leftarrow t_2, ..., x_i \leftarrow t_i\}$ as the substitution of the terms $t_1, t_2, ..., t_i$ for variables $x_1, x_2, ..., x_i$ in a formula $f \in wff(\mathcal{L}_O)$. Thus, $\Theta(f(x_1, x_2, ..., x_i)) \equiv f(t_1, t_2, ..., t_i)$. We will denote as $\vartheta_{(wff(\mathcal{L}_O),\mathcal{S})}$ the set of all possible substitution instances containing the variables in $wff(\mathcal{L}_O)$ and the terms in $\mathcal{S}$.

Definition 1 (Norm) *A 'norm' n is a tuple $n = \langle f_A, f_M, f_D, f_w, w\rangle$, where*

- $f_A, f_M, f_D, f_w \in wff(\mathcal{L}_O)$, $w \in O$,
- f_A, f_M, f_D *respectively represent the activation, maintenance, and deactivation conditions of the norm; f_w is the explicit representation of the target of the norm, and w is the subject of the norm (role or agent).*

We can formalise the norms of Definition 1 as the equivalent deontic expression (using the formalism of [7]):

Property 1 *A norm is considered fulfilled if, and only if:*

$$f_A \rightarrow [O_w(E_w f_w \leq \neg f_M)\,\mathcal{U}\, f_D]$$

where U is the CTL* until operator.

Intuitively, Property 1 states that after the norm activation, the subject is obliged to see to it that the target becomes true before the maintenance condition is negated (either the deadline is reached or some other condition is broken) until the norm is deactivated (which is either when the norm is fulfilled or has otherwise expired).

Definition 2 (Violation handling norm[3]) *A norm $n' = \langle f'_A, f'_M, f'_D, f'_w, w'\rangle$ is a violation handling norm of $n = \langle f_A, f_M, f_D, f_w, w\rangle$, denoted as $n \rightsquigarrow n'$ iff $f_A \wedge \neg(f_M \mathcal{U} f_D) \vdash f'_A$*

Violation handling norms are special in the sense that they are only activated once another norm is violated. They are used as *sanctioning norms*, if they are to be fulfilled by the norm violating actor (e.g., the obligation to pay a fine if the driver broke a traffic sign), or as *reparation norms*, if they are to be fulfilled by an institutional actor (e.g. the obligation of the authorities to fix the broken traffic sign).

A norm is defined in an abstract manner, affecting all possible participants enacting a given role. Whenever a norm is active, we will say that there is a *norm instance* $ni = \langle n, \theta\rangle$ for a particular norm n and a substitution instance Θ.

In order to track the normative state of an institution at any given point of time, we will define three sets: an instantiation set IS, a fulfillment set FS, a violation set VS, and

[3] Informally: the unfulfillment of the obligation of norm n entails the activation of norm n'.

Event processed:

$$\frac{e_i = \langle \alpha, t, p \rangle}{\langle s \rangle \overset{ep}{\triangleright} \langle s \cup \{p\} \rangle} \tag{1}$$

Norm instantiation:

$$\frac{activated(n, \Theta) \qquad n \in N \qquad \neg \exists n' \in N, n' \rightsquigarrow n \qquad \langle n, \Theta \rangle \notin is}{\langle is \rangle \overset{nii}{\triangleright} \langle is \cup \{\langle n, \Theta \rangle\} \rangle} \tag{2}$$

Norm instance violation:

$$\frac{\neg maintained(\langle n, \Theta \rangle) \qquad NR = \bigcup_{n \rightsquigarrow n'} \langle n', \Theta \rangle \qquad n \in N \qquad \langle n, \Theta \rangle \in is \qquad \langle n, \Theta \rangle \notin vs}{\langle is, vs \rangle \overset{niv}{\triangleright} \langle (is - \{\langle n, \Theta \rangle\}) \cup NR, vs \cup \{\langle n, \Theta \rangle\} \rangle} \tag{3}$$

Norm instance fulfilled:

$$\frac{deactivated(n, \Theta') \qquad n \in N \qquad \langle n, \Theta \rangle \in is \qquad \Theta' \subseteq \Theta}{\langle is, fs \rangle \overset{nif}{\triangleright} \langle is - \{\langle n, \Theta \rangle\}, fs \cup \langle n, \Theta \rangle \rangle} \tag{4}$$

Norm instance violation repaired:

$$\frac{\langle n', \Theta \rangle \in fs \qquad n, n' \in N \qquad n \rightsquigarrow n' \qquad \langle n, \Theta \rangle \in vs}{\langle vs, rs \rangle \overset{nir}{\triangleright} \langle vs - \{\langle n, \Theta \rangle\}, rs \cup \{\langle \langle n, \Theta \rangle, \langle n', \Theta \rangle \rangle\} \rangle} \tag{5}$$

Figure 1. Inference rules for the transition relation $\triangleright$

a repairment set RS. Each of them contains norm instances $\{\langle n_i, \Theta_j \rangle, ..., \langle n_{i'}, \Theta_{j'} \rangle\}$. We adapt the semantics for normative states from [11]:

Definition 3 (Norm Lifecycle) *Let $ni = \langle n, \Theta \rangle$ be a norm instance, such that $n = \langle f_A, f_M, f_D, w \rangle$, and s be a state of the world with an expansion $F(s)$. Then we define the lifecycle for a norm instance ni by the following normative state predicates:*

$activated(ni) \Leftrightarrow \exists f \in F(s), \Theta(f_A) \equiv f$
$maintained(ni) \Leftrightarrow \exists \Theta', \exists f \in F(s), \Theta'(f_M) \equiv f \wedge \Theta' \subseteq \Theta$
$deactivated(ni) \Leftrightarrow \exists \Theta', \exists f \in F(s), \Theta'(f_D) \equiv f \wedge \Theta' \subseteq \Theta$
$instantiated(ni) \Leftrightarrow ni \in IS$
$violated(ni) \Leftrightarrow ni \in VS$
$fulfilled(ni) \Leftrightarrow ni \in FS$
$repaired(ni, ni') \Leftrightarrow \langle ni, ni' \rangle \in RS$

where IS is the instantiation set, FS is the fulfillment set, VS is the violation set, and RS is the set of those norm instances ni' that have repaired a norm instance ni.

Definition 4 (Event) *An event e is a tuple $e = \langle \alpha, t, p \rangle$, where*

- *$\alpha \in O$, an actor of the system,*
- *t is the timestamp of the reception of the event, and*
- *given a fully grounded subset of the set of states of the world $p' \in S : p = p' \vee p = \neg p'$*

We define E as the set of all possible events, $E = \mathcal{P}(P \times S)$.

Definition 5 (Normative Monitor) *A Normative Monitor M_N for a set of norms N is a tuple $M_N = \langle N, S, IS, VS, FS, RS, E \rangle$.*

Γ_{M_N} is the set of all possible configurations of a Normative Monitor M_N.

Definition 6 (Labelled Transition System) *The* Labelled Transition System LTS_{M_N} *for a Normative Monitor* M_N *is defined by* $LTS_{M_N} = \langle \Gamma_{M_N}, L, \rhd \rangle$ *where*

- $L = \{ep, nii, niv, nif, nir\}$ *is a set of labels, respectively representing* event processed*,* norm instantiation*,* norm instance violation*,* norm instance fulfilled*, and* norm instance violation repaired*, and*
- $\rhd$ *is a* transition relation *such that* $\rhd \subseteq \Gamma_{M_N} \times L \times \Gamma_{M_N}$

The inference rules for the transition relation $\rhd$ are described in Figure 1.

This formalism, as shown in [2], has been reduced to the semantics of general production systems and an implementation in DROOLS is already available.

3. Formalizing Norm dependency

In this section we extend the monitoring formalism introduced in Section 2 for defining dependencies between formulas. Then, these dependencies are then extended to dependencies between norms and monitors.

3.1. Inter-formula dependency

Informally speaking, two formulas are dependent when they share common predicates and, therefore, some parts of the state of the world may affect them both at the same time. More formally, we define two formulas as mutually dependent if, and only if, there is at least one possible state of the world in $\mathcal{S}$ at which both formulas have a grounding of their predicates and both groundings share one or more predicates.

In order to formalise this, we introduce the transition function δ which given a formula and the actual state of the world will output a not partial grounding:

Definition 7 (State grounding function) *We define the function* δ *as:*

$$\delta : wff(\mathcal{L}_O) \times \mathcal{S} \rightarrow \vartheta_{(wff(\mathcal{L}_O),\mathcal{S})}$$

Given $f \in wff(\mathcal{L}_O)$ *and* $\sigma \in \mathcal{S}$*:*

$$\delta(f, \sigma) = \begin{cases} \Theta & \textit{iff } \exists \Theta : \exists g \in F(\sigma) : \Theta(f) \equiv g \\ \emptyset & \textit{otherwise} \end{cases}$$

Using the transition function δ we can state that two formulas are dependent if, given the same state of the world, it returns two sets of substitutions –groundings– with at least one common element:

Definition 8 (Formula dependency) *Given two formulas* f_x*,* f_y *such that* $f_x, f_y \in wff(\mathcal{L}_O)$ *we state* f_x *and* f_y *are dependent, and denote it by* $f_x \leftrightsquigarrow f_y$

$$f_x \leftrightsquigarrow f_y \Leftrightarrow \exists \sigma \in \mathcal{S} : \delta(f_x, \sigma) \cap \delta(f_y, \sigma) \neq \emptyset$$

The previous definitions do not take into consideration how the formulas are defined and thus are too general. In our particular case, we use formulas in propositional logic which are normalisable to DNF. The following definitions introduce how to compute $f_x \nleftrightsquigarrow f_y$ with DNF formulas.

Given a formula f, we define $\mathcal{D}(f)$ as the DNF of f. The DNF is the disjunction of a set of conjunctive clauses. Given a conjunctive clause f', the function $\mathcal{C}(f')$ returns the set of predicates p on the clause. Formally: $\mathcal{C}(f') = \bigcup_{p \in f'} p$. We use $\mathcal{C}(f')$ to define the function $\mathcal{F}(f)$ returning the predicates of a formula. The set of predicates of a formula in DNF form is the union of the set of predicates of every conjunctive clause f' in the formula:

Definition 9 (Predicates of a formula) *Given the DNF form of a formula $f = f'_1 \vee f'_2 \vee ... \vee f'_n$, the function $\mathcal{F}(f)$ is defined as follows:*

$$\mathcal{F}(f) = \bigcup_{f' \in \mathcal{D}(f)} \mathcal{C}(f')$$

For instance, let f_M be a formula that defines when an agent is working either on the camera ready version or on the presentation of a paper he has submitted to a conference:

$$\begin{aligned} f_M = (&camera\_ready\_of(R, P) \wedge submited\_to(P, C) \wedge working\_on(A, R)) \vee \\ &presentation\_of(S, P) \wedge submited\_to(P, C) \wedge working\_on(A, S)) \\ \mathcal{F}(f_M) = \{&camera\_ready\_of(R, P), presentation\_of(S, P), submited\_to(P, C), \\ &working\_on(A, R), working\_on(A, S)\} \end{aligned}$$

Intuitively, we can see that given two formulas f, f', the fact that f and f' have at least one common predicate is equivalent to the fact that they are dependent:

Proposition 1 *Given two formulas f, f':*

$$\mathcal{F}(f) \cap \mathcal{F}(f') \neq \emptyset \Leftrightarrow f \nleftrightsquigarrow f'$$

In this section, the concepts of *predicate of a clause* and *predicate of a formula* have been used for introducing the concept of formula dependency. This last concept is used in the next section for introducing the concept of norm dependency.

3.2. Inter-norm dependency

This subsection extends the definitions provided in *Section* 3.1 to provide a formal definition of dependencies between norms.

First, we use $\mathcal{F}(f)$ to define the function $\mathcal{Q}(n)$ that returns the predicates of a norm:

Definition 10 (Predicates of a Norm) *Given a norm $n = \langle f_A, f_M, f_D, f_w, w \rangle$, the function $\mathcal{Q}(n)$ is defined as follows:*

$$\mathcal{Q}(n) = \bigcup_{f \in \{f_A, f_M, f_D\}} \mathcal{F}(f)$$

The concept of *predicates of a norm* allows us to introduce the concept of norm dependency. We state two norms are dependent if their sets of predicates are not disjoint:

Definition 11 (Norm Dependency) *Given two norms n and n' we state they are dependent, and denote it by $n \leftrightarrows n'$ when:*

$$n \leftrightarrows n' \iff \mathcal{Q}(n) \cap \mathcal{Q}(n') \neq \emptyset$$

The concept of norm-dependency is used to define the concept of normative monitor dependency. Two normative monitors are dependent if and only if for at least one norm in one of the monitors, there is a norm on the other monitor dependent on it:

Definition 12 (Monitor Dependency) *Given two monitors M_N and $M_{N'}$ we state they are dependent, and denote it by $M_N \leftrightarrows M_{N'}$ when:*

$$M_N \leftrightarrows M_{N'} \iff \exists n \in N, \exists n' \in N' : \mathcal{Q}(n) \cap \mathcal{Q}(n') \neq \emptyset$$

4. Architecture for distributed monitoring

This section introduces the architecture for splitting a normative context –bound to a single monitor– to a set of normative contexts bound to a set of interconnected and distributed monitors. The idea is to perform the context splitting in such a way that the connections between distributed monitors are reduced to the minimum. This will allow to reduce communication overhead between monitors and to reach higher efficiency.

4.1. From norms to graphs

This subsection introduces the idea of modelling a set of norms as a graph. First of all, we define the graph resulting from a set of norms. Then we introduce some basic concepts on *Strongly Connected Component* (*SCC* from now on) applied to graphs. Finally, we glue all these concepts together by applying SCCs [4] to sets of norms.

Definition 13 (Graph) *We define a (directed) graph as a pair $G = \langle V, E \rangle$. V is a finite set of nodes and $E \subseteq V \times V$ is a set of edges. We denote the existence of a path from v to w of length k as $path(v, w, k)$.*

From the definition of a graph we can model the normative framework as a graph:

Definition 14 *Given a monitor for a normative framework M_N we define a graph $G_N = \langle G_V, G_E \rangle$ such that:*

1. *The set of nodes in the graph is the set of norms: $G_V = N$*
2. *$\forall_{n,n'} \in N \ : \ (n, n') \in G_E \Leftrightarrow n \leftrightarrows n'$*

G_N denotes the graph representation of the norms in the normative monitor M_N.

Now we are ready to apply our normative model to *Strongly Connected Components*. A *SCC* of $G = \langle V, E \rangle$ is a maximal set $C \subseteq V$ such that: $\forall v, w \in C, \ path(v, w, i)$. A *SCC* is not trivial if: $\forall v, w \in C, \ path(v, w, i) \wedge i > 0$. Given a node $v \in V$, $SCC(v)$ denotes the SCC that contains v.

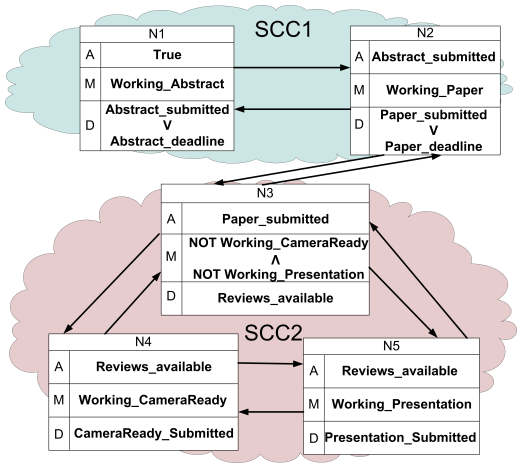


Figure 2. Graph with norm dependencies and resulting strongly connected components(SCC)

Definition 15 (Component graph of a normative context) *The component graph* $\mathcal{G}_N$ *of the normative context* G_N *is* $\mathcal{G}_{\mathcal{N}}(G_N) = \langle V', E' \rangle$*, where*

- $V' = \{SCC(v) \mid v \in V\}$
- $E' = \{(C = SCC(v), C' = SCC(v')) \mid C \neq C' \wedge (v, v') \in E\}$

Figure 2 shows a set of norms and dependency relationships among them. *SCC* are identified and depicted as clouds grouping the norms.

4.2. Distributing the graph

In the previous section we have introduced means for representing the normative context associated to a monitor as a Graph, and computing the *SCC* of the graph. The idea is replacing every SCC by a single vertex, in order to obtain a smaller graph, known as *component graph*. We can reduce the original problem to sub-problems on each particular SCC, plus one more sub-problem on the *component graph*. If such sub-problems are distributed among different computational nodes we have effectively distributed the computational process associated to the original problem.

When applying the notion of graph and *SCC* to a Normative Monitor for a set of norms, we can effectively reduce the problem of monitoring the original set of norms to several sub-problems: monitoring the sub-set of norms in each *SCC* of the graph. Each of these sub-problems can be computed by a different normative monitor. The process of splitting a Normative Monitor consists of three steps: 1) Computing the graph associated to the set of norms of the monitor. Then, computing the *component graph* of the graph. 2) For every *SCC* (that is, every node in the *component graph*), create an empty monitor that will compute these sub-set of norms from now on. 3) For every monitor created, subscribe it to another monitor[4] when the corresponding *SCC* are connected by a vertex on the *component graph* 14 vertices are created based on norm dependency. Therefore

[4]This subscription allows a monitor to notify events to other monitors dependent on it

Automatic learning of preferences in numeric criteria

Lucas MARIN [a,1], Antonio MORENO [a], David ISERN [a]
[a] *Universitat Rovira i Virgili, Departament d'Enginyeria Informàtica i Matemàtiques, ITAKA Research Group, Tarragona (Catalunya)*

Abstract. Due to the astonishing speed at which new content is created and published on the Web, it is increasingly difficult for users to make the most appropriate decisions in front of an overwhelming amount of information. Recommender systems try to help users by analyzing and ranking the available alternatives according to their preferences and interests, modeled in user profiles. One important problem to solve in the development of these systems is how to discover the user preferences, and how to maintain them dynamically. In this work we propose to use the information given by a user in his/her interaction with the recommender system (*e.g.* the selection of the news to be read every morning) to infer his/her preferences on several criteria on which the decision alternatives are defined. More specifically, the paper is focused in learning the most preferred value for the user in the case of numerical attributes. A methodology to adapt the user profile in a dynamic and automatic way is presented. The adaptations may be performed after each interaction of the user or after the system has gathered enough information from several user selections. We have developed a framework for the automatic evaluation of the performance of the adaptation algorithm that permits to analyze the influence of different parameters. The obtained results show that the adaptation algorithm is able to learn a very accurate model of the user preferences after a certain amount of interactions.

Keywords. Recommender systems, profile adaptation, preference learning

Introduction

A recommender system (RS) assists and augments the natural social process of making choices without sufficient personal experience of the alternatives [3]. Every suggestion obtained from using a RS is related to a decision-making process, such as what travel to make or what online news to read. Nowadays, this kind of systems is used to personalize services through e-commerce and advertising.

RSs take into account the interests of the user to rank a set of possible solutions to a decision-making problem. The main goals of RSs are to *measure* and *predict* the most suitable alternative for her/him. To achieve these goals, two of the most challenging tasks are the representation and the management of the user interests through a user profile. A user profile represents preferences about a set of criteria by which the alternatives of the recommendation problem are evaluated. As pointed out in [1], the design of recommender systems which evaluate the alternatives through a multiple

[1] Corresponding Author: Universitat Rovira i Virgili, ITAKA Research Group. Av. Països Catalans, 26, 43007 Tarragona (Catalunya); E-mails: lucas.marin@urv.cat, antonio.moreno@urv.cat, david.isern@urv.cat

criteria approach is a scarcely explored area, and the few tests conducted in this area have proven that the use of multi attribute ratings improves the recommendation accuracy. In the literature, the reader can find different approaches. For instance, in [14] the user profile contains a set of words and the probabilities with which they are interesting for the user, which are used to rank web sites before recommending them. In [8], the authors use a Bayesian network to represent relationships between words which model different types of users, and in [12], the preferences are modeled using fuzzy linguistic variables.

The first task of the management of the user preferences is the creation of an initial profile. As discussed in [11], the initial profile can be empty, can be deduced from a training data set, can be manually initialized (*e.g.* via a questionnaire), can be learnt by using machine learning techniques, or can be initialized using a predefined set of stereotypes. As pointed out in [3,15], most recommender techniques suffer from what is called the "cold-start problem", meaning that until a certain amount of users have used the system and it has learnt from their feedback, no reliable recommendations can be obtained. As it has been analyzed in [10,11], the dynamic adaptation of user preferences requires some kind of relevance feedback, which can be acquired explicitly, implicitly, and combining these two approaches.

Explicit feedback is obtained when users are required to evaluate items, indicating how relevant or interesting an item is to the user in a numeric or linguistic scale. These systems, such as [12,13], offer high performance and simplicity. However, explicit feedback is sometimes not desirable because it has some serious limitations. One of them is that numeric scales may not be adequate for describing the reactions humans have to items. Moreover, as studied in [15], users are usually reluctant to spend time giving explicit feedback and only 15% of the users would supply it even if they were encouraged to do so. On the contrary, implicit feedback is obtained by monitoring the user actions and automatically inferring the user preferences. The amount of collected data is consequently very large, the computation needed to derive recommendations for adaptations is extensive, and the confidence in the suitability of these adaptations is likely to be relatively low. This approach has been less explored, although some existing methods (*e.g.*, [2,5]) have shown promising results. As an example, the authors of [7] present techniques to improve web search by using implicit feedback extracted from the users interaction (*e.g.*, the position of the link the user clicks among the results of the search). The work conducted in this paper proposes the use of an implicit feedback by observing the selections made by the user from a rated set of alternatives.

With regards to the focus of this paper, which is the adaptation of the user profile from the interaction of the user with the system, a recent and relevant work in this area can be found in [4]. The authors present a fuzzy framework in which the resources to recommend are tagged with metadata that describe the most prominent features of each resource. The user profile describes, using fuzzy sets, the preferences of the user on a set of criteria. A matching mechanism that uses the membership functions associated with the user profile and the resources permits the evaluation of the similarity of both elements, recommending the most similar item. In addition, an adaptation mechanism helps to update the user profile by taking into account the features of the selected resource. However, one of the drawbacks of this framework is the need for tagged information attached to each resource. This implies a subjective annotation of all the resources, taking into account the available criteria.

As studied in [9], most of the RSs omit that the user interests can change over time. This work proposes a combination of two modules: a short-term and a long-term user

profile. The changes are proposed taking into account the last action done by the user (short-term) combined with a past history of actions done (long-term). In this work, focused in suggesting Web pages, the user profile is represented through a taxonomy of terms labeled with probabilities, which evolve over time.

The rest of the paper is organized as follows. Section 1 provides some preliminaries, introducing the data representation structures and defining our approach of RS. Section 2 explains the numerical adaptation techniques designed, which are then tested and evaluated. Finally, Section 4 gives the main conclusions of the paper and suggests some lines of future research.

1. Preliminaries

This section describes a set of concepts required before the introduction of the adaptation algorithm. The first section explains how the set of alternatives and the user profile are represented. The second one describes how it is possible to compute the similarity between the user profile and each of the alternatives, in order to recommend the alternatives that fit better the user preferences.

1.1. Data representation

The group of alternatives of a RS (A) constitutes the set of possible solutions to the recommendation problem in which the user requires aid, which is evaluated by a set of points of view or criteria. Considering a set of criteria $C = \{c_1, c_2, \dots, c_k\}$, an alternative $a \in A$ can be defined *as* $a = \{a_1, a_2, \dots, a_k\}$, where a_i is the value of the criterion c_i in the alternative a. The alternatives considered in this paper are formed by numerical continuous attributes (real numbers). Each of those criteria has its own bounds, $range_{c_i} = (c_i^{min}, c_i^{max})$, where c_i^{min} and c_i^{max} are the minimum and maximum values of the criterion c_i, respectively.

The preferences of the user need to be represented in a profile in order to be employed by the RS to perform personalized recommendations. The user profile stores a value of preference for each criterion, that is, the numeric value for which the user has the greatest interest. Thus, the profile P for the user u is defined as $P_u = \{v_{pref_1}, v_{pref_2}, \dots, v_{pref_k}\}$, where v_{pref_i} represents the value of maximum preference over the criterion c_i. In fact, the profile in this approach can be seen as another alternative which has the ideal values in each criterion.

1.2. Recommendation process

Any RS starts its recommending process by evaluating all the possible solutions that form the set of alternatives of the recommendation problem. In our approach, that evaluation is conducted by calculating the distance of each alternative to the stored profile. The distance between two alternatives (the profile can also be seen as an alternative for this purpose, as pointed out in the previous section) is calculated with the following function:

$$dist(a, b) = \frac{1}{k} \sum_{i=1}^{k} \frac{|a_i - b_i|}{c_i^{max} - c_i^{min}} \quad (1)$$

In this expression a_i and b_i are the values for the criterion i in the alternatives a and b, respectively, and k is the number of criteria. The distance is the average of the normalized differences of each component, because all criteria are assumed to be independent and with the same weight in the decision making process.

After all the distances have been calculated, the RS sorts the alternatives by that distance value in increasing order, meaning that the first ranked alternative is the one with a lower distance to the current stored profile, and the one that would better suit the user interests.

2. Numerical adaptation

After having ranked the alternatives, the process of adapting the numerical attributes of the user profile is conducted on two consecutive phases, as represented in Figure 1. The first one involves gathering user information about his/her interests (relevance feedback) in order to extract enough information to adapt the user preferences in the profile. Then, the adaptation algorithm investigates which changes in the profile have to be done.

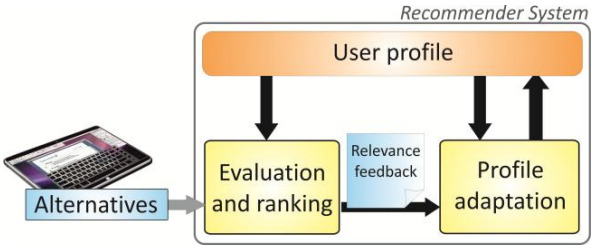


Figure 1. Recommender system architecture

2.1. Relevance feedback

In order to adapt the user profile it is necessary to obtain some information about the user interaction with the RS. This information, as explained in the introduction, is called relevance feedback and can be extracted explicitly or implicitly.

Our approach in this paper uses implicit feedback to obtain information about the user interests since it is less intrusive and requires no additional effort from the user. When the user asks for a recommendation and finally selects one option as his/her favorite, two pieces of implicit feedback are extracted from this action: the alternative the user selected and a set of alternatives that were incorrectly ranked by the system above the selection called over ranked alternatives.

2.2. Adaptation algorithm

The numeric adaptation of the user profile presented in this paper is inspired by Coulomb's Law, which states that "the magnitude of the electrostatic force of interaction between two point charges is directly proportional to the scalar multiplication of the magnitudes of the charges and inversely proportional to the square of the distances between them". The main idea is to consider the preference value

stored in the profile for each criterion as a charge with the same polarity as the values of the same criterion on the over ranked alternatives, and with opposite polarity to the criterion value in the selected alternative. In that way, the value of the profile is pushed away by the values in the over ranked alternatives and, at the same time, attracted by the value in the selected alternative.

Two stages have been considered in the adaptation algorithm. The first one, called on-line adaptation process, is performed each time the user interacts with the platform, *i.e.*, each time the user asks for a recommendation and receives a response. The other stage, called off-line process, is performed after a certain amount of interactions with the user.

2.2.1. On-line adaptation process

The information available for each execution of the adaptation process is the selection of the user and the over ranked alternatives that were better ranked by the system. The objective in this stage of the adaptation is to extract enough evidence from this information to make a change in the profile. In order to do it, it is necessary to have a minimum number of over ranked alternatives (*mo*). Otherwise, the extracted evidences would not be reliable enough.

In order to calculate the change of the value of preference in the user profile of a certain criterion *j*, it is necessary to study the attraction forces over each criteria done by the selected alternative ($F^s = \{F_1{}^s, F_2{}^s, \dots, F_k{}^s\}$) and the repulsion force done by the over ranked alternatives ($F^o = \{F_1{}^o, F_2{}^o, \dots, F_k{}^o\}$), as represented in the example in Figure 2, in which the *j*-th value of the five over ranked alternatives o^0, o^1, o^2, o^3 and o^4 cause a repulsion force F_j^o, and the value for the same criterion of the selected alternative s causes an attraction force F_j^s, both forces over the *j*-th value of the profile p.

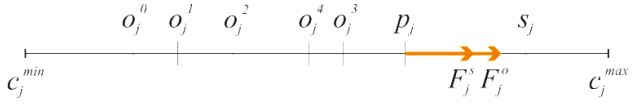


Figure 2. Attraction and repulsion forces

The force done by the selected alternative is defined as:

$$F_j^s = \left(c_j^{max} - c_j^{min}\right) \frac{1}{\left|s_j - p_j\right|^{\alpha}} * \frac{s_j - p_j}{\left|s_j - p_j\right|}, \tag{2}$$

where c_j^{min} and c_j^{max} are the minimum and maximum values of criterion j, s_j is the value of the criterion j in the selected alternative and p_j is the value of the same criterion in the stored profile.

The force applied by the over ranked alternatives is defined as:

$$F_j^o = \left(c_j^{max} - c_j^{min}\right) \sum_{i=1}^{no} \left(\frac{1}{\left|p_j - o_j^i\right|^{\alpha}} * \frac{p_j - o_j^i}{\left|p_j - o_j^i\right|} \right), \tag{3}$$

where o_j^i is the value of the criterion j in the over ranked alternative i, and *no* is the number of over ranked alternatives. The influence of the parameter α has been empirically evaluated and the tests that have been conducted indicate that the value with which the best results were obtained is $\alpha = 3/4$. Greater values for α caused F^o

to be too low while lower values caused F^o to be too high, compromising the adequate balance between the attraction and repulsion forces.

Finally, both forces are summed up and the resulting force is calculated as $F_j^{on-line} = F_j^o + \theta * F_j^s$ where θ is a correction factor of the attraction force of the selected alternative. The constant θ is calculated as $n * \beta$, where n is the total number of alternatives and β is a parameter that adjusts the sum of the forces in a similar way as the parameter α. This correction factor permits to tune the importance of the selected alternative in contrast with the over ranked ones. Empirical tests on the repercussion of the parameter β have indicated that the best results were obtained using a value of $\beta = 7/4$.

When the number of over ranked alternatives does not reach a minimum value (*mo*), the process does not make any changes in the profile and the current over ranked alternatives are stored for the next execution of the off-line adaptation process. This required number of over ranked alternatives is based in the fact that if we have very few over ranked alternatives, the possible changes in the profile are deduced from a small amount of information, making the whole reasoning process unreliable.

2.2.2. Off-line adaptation process

The techniques designed for the on-line adaptation process do not find user trends over time since they are just based in the observation of a single selection, and that is what the off-line adaptation process intends to improve. Gathering information through the observation of more than a single interaction allows having a larger amount of useful data to decide which changes to do in the profile.

The off-line adaptation process is triggered in two ways, which are independent between them, and can be executed at the same time. The first one is in charge of evaluating the selections the user makes while the second one is responsible of managing the over ranked alternatives the user discards when he/she asks for a recommendation. The suitability of execution of the off-line process (in any of its two possible variants) is checked after each recommendation.

User selections are stored for the off-line adaptation process and, after a certain minimum amount of selections have been gathered (*rs*), the off-line adaptation process is triggered. In this process it is necessary to calculate only the attraction forces done by the stored selected alternatives (F^s), which is:

$$F_j^s = \left(c_j^{max} - c_j^{min}\right) \sum_{i=1}^{ns} \left(\frac{1}{\left|s_j^i - p_j\right|^{\alpha}} * \frac{s_j^i - p_j}{\left|s_j^i - p_j\right|} \right), \tag{4}$$

where s_j^i is the value of the criterion j in the selected alternative i and *ns* is the number of stored selections. The forces calculated with this kind of execution of the off-line adaptation process are identified as $F^{off-line_1} = \{F_1^{off-line_1}, \ldots, F_j^{off-line_1}\}$, where $F_i^{off-line_1} = F_i^s$.

Moreover, as indicated in the previous sub-section, when the on-line process does not have enough over ranked alternatives to produce any change in the profile, they are stored. When the accumulated stored over-ranked alternatives reach a certain minimum number (which is the same *mo* used in the on-line adaptation process), the off-line adaptation process is triggered to calculate the forces over the profile values done by those alternatives (F^o), which are calculated using Eq. (3). The forces calculated with

this kind of execution of the off-line adaptation process are identified as $F^{off-line_2} = \{F_1^{off-line_2}, \dots, F_j^{off-line_2}\}$, where $F_i^{off-line_2} = F_i^o$.

3. Evaluation

The evaluation has been conducted in a domain where the user asks for a recommendation over a set of alternatives, based in a certain set of numerical criteria. A set of 1500 alternatives, which have been generated randomly with a uniform distribution, has been used for the purpose of this evaluation. The set of criteria used to define each alternative is composed by five attributes whose possible values range from 0 to 100.

The whole evaluation process is divided in two parts: recommendation and adaptation. In the first one, the alternatives are evaluated and rated, while in the second the user profile is updated according to the selections. In the final part of this section, the obtained results are discussed.

3.1. Recommendation step

The first step in the evaluation process consists in separating the alternatives in blocks of 15 elements, generating 100 different recommendation problems. At each iteration of the evaluation process, only one block of 15 alternatives is considered.

The recommendation step starts by evaluating all of the 15 alternatives that form a block of alternatives using Eq. (1) as explained in section 1.2. As a result, we obtain the list of 15 alternatives ranked according to the distance to the user's current profile.

After the rating and ranking step, a user interacting with the platform would select his/her favorite alternative. In the evaluation we have simulated that step by considering an ideal profile created manually, which is the profile we want the current profile to tend to. The ideal profile stores the preferred value of the user for each of the criteria. We consider as user selection the alternative with a lowest distance to that ideal profile. The alternatives that were ranked above this one are the set of over ranked alternatives that, along with the selected alternative, form the information required for the adaptation process.

3.2. Adaptation step

When the selection and the over ranked alternatives have been identified, the adaptation step takes that information as input in order to decide which changes to be made in the profile, as explained in Section 2.2.

After the block of 15 alternatives has been evaluated and the relevance feedback extracted, the adaptation step calculates the forces used to adapt the profile in the on-line process (Eqs. (2) and (3)) and, if it is necessary to run the off-line adaptation process, obtains the adaptation forces with Eqs. (3) and (4). When the stored profile has been modified according to the results of the adaptation processes, the evaluation process gathers the next block of alternatives and starts again the evaluation process.

3.3. Results

The analysis of the results obtained in the evaluation of the algorithms has been conducted from two points of view. First, an analysis of the evolution of the distances between the ideal and the adapted profile is conducted, using only the on-line process or both the on-line and off-line processes. Afterwards, an evaluation of the performance of the RS has been conducted, studying the position of the selected alternative on the sorted alternatives from the recommender. Three tests with three different random initial profiles have been conducted in order to test our approach and the results included in this section are computed by calculating the average of said three tests.

The graphical representation in Figure 3 shows the performance of the adaptation algorithm by measuring the distance between the ideal profile (the profile that we aim to reach with the initial profile) and the adapted profile (the initial randomly created profile which is being adapted through the evaluation process).

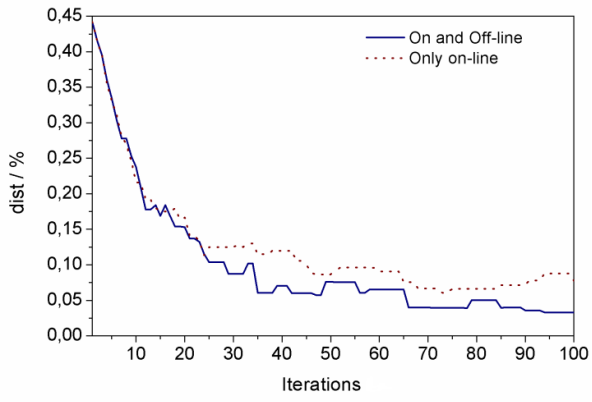


Figure 3. Distances between the ideal and the adapted profiles using the adaptation processes

To understand the meaning of the distance values in this graph, it is necessary to observe that the initial distance of 0.41 means that the stored preferred value in the profile for each criterion is, in average, at a distance to the ideal value of 41% of the range of the domain for that criterion. This means that if the domain of a certain criterion is of 100 units, the initial distance between the profile and ideal values is 41 units.

Considering that, we can say that it is better to use both the on-line and off-line processes since from near the 25th iteration of the evaluation process it can be seen that by using them both we reach a distance around 0.04 between profiles, obtaining a final distance of 0.05. In other words, each adapted value in the profile is, in average, at a distance of 5% of the domain range to its corresponding ideal value. It is also worth mentioning that the best results are obtained from the 50th interaction and no important improvement in the distances is experienced past this point.

Figure 4 shows in which position of the ranking performed by the RS the user selection is located. It can be seen how, at the first interactions, the selected alternative is not very well ranked by the RS, since the user selections are found anywhere in the resulting ranking. When observing Figure 3 we said that near the 50th interaction it was the point from which the profile was best adapted. This fact can also be observed in

Figure 4 by noting that from this same point (from near the 50th interaction) the user selection is normally ranked among the first three alternatives, and in the great majority of cases, it is ranked in the first position.

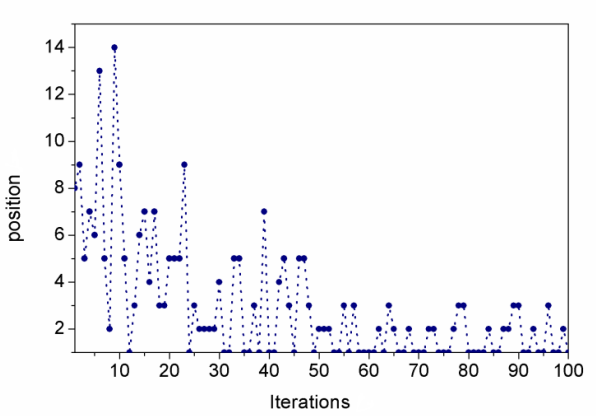


Figure 4. Ranking position of the favorite alternative in the recommendation result

4. Conclusions and future work

The adaptation algorithms defined in this paper have been shown in the previous section to correctly adapt the preferences of a user over time. It has been seen that, after fifty interactions with the user, the profile was accurate enough to rank the most interesting item for the user among the first three positions in the majority of cases. Thus, we can argue that the algorithms seem suitable for being used in frequent tasks such as recommending "what TV program to watch" or "what film to see in the cinema" rather than "what car to buy". That issue, however, can be overcome by including collaborative filtering techniques in the RS such as sharing the same profile between a set of similar users. This would cause the adaptation to be quicker due to the fact that more information is gathered in less time since multiple users are interacting with the platform at the same time with the same profile.

Another important improvement of this work would be to join the work presented in this paper with the approach presented in [10] in order to allow recommendations in a domain where alternatives are defined by both linguistic [6] and numeric criteria.

5. Acknowledgements

This work has been partially supported by the Universitat Rovira i Virgili (a pre-doctoral grant for L. Marin), the Spanish Ministry of Science and Innovation (DAMASK project, *Data mining algorithms with semantic knowledge*, TIN2009-11005) and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan).

References

[1] G. Adomavicius, K. YoungOk, New Recommendation Techniques for Multicriteria Rating Systems, *IEEE Intelligent Systems* **22** (3) (2007), 48-55.

[2] L. Baltrunas, X. Amatriain, Towards Time-Dependant Recommendation based on Implicit Feedback, in: *Workshop on Context-aware Recommender Systems in conjunction with the 3rd ACM Conference on Recommender Systems, RecSys 09*, G. Adomavicius, F. Ricci, eds, New York, USA, 2009, 25-30.

[3] R. Burke, Hybrid recommender systems: survey and experiments, *User Modeling and User-Adapted Interaction* **12** (4) (2002), 331–370.

[4] G. Castellano, C. Castiello, D. Dell'Agnello, A. M. Fanelli, C. Mencar, M. A. Torsello, Learning Fuzzy User Profiles for Resource Recommendation, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **18** (4) (2010), 389-410.

[5] E. Cheng, F. Jing, M. Li, W. Ma, H. Jin, Using Implicit Relevance Feedback to Advance Web Image Search, in: *IEEE International Conference on Multimedia and Expo, ICME 2006*, IEEE Computer Society, Toronto, ON, Canada 2006, 1773-1776.

[6] D. Isern, L. Marin, A. Valls, A. Moreno, The Unbalanced Linguistic Ordered Weighted Averaging Operator, in: *IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2010*, IEEE Computer Society, Barcelona, Catalonia, 2010, 3063-3070.

[7] T. Joachims, F. Radlinski, Search Engines that Learn from Implicit Feedback, *Computer* **40** (8) (2007), 34-40.

[8] K.-Y. Jung, K.-W. Rim, J.-H. Lee, Automatic Preference Mining through Learning User Profile with Extracted Information, in: *Structural, Syntactic, and Statistical Pattern Recognition Joint IAPR International Workshops, SSPR 2004 and SPR 2004*, A. Fred, T. Caelli, R. Duin, A. Campilho, D. de Ridder, eds, Springer Berlin / Heidelberg, Lisbon, Portugal, 2004, 815-823.

[9] L. Li, Z. Yang, B. Wang, M. Kitsuregawa, Dynamic adaptation strategies for long-term and short-term user profile to personalize search, in: *Joint 9th Asia-Pacific Web Conference, APWeb 2007, and 8th International Conference, on Web-Age Information Management, WAIM 2007, Advances in Data and Web Management*, G. Dong, X. Lin, W. Wang, Y. Yang, J. X. Yu, eds, Springer Verlag, Huang Shan, China, 2007, 228-240.

[10] L. Marin, D. Isern, A. Moreno, A Generic User Profile Adaptation Framework, in: *13th International Conference of the Catalan Association for Artificial Intelligence, CCIA 2010*, R. Alquézar, A. Moreno, J. Aguilar-Martin, eds, IOS Press, l'Espluga de Francolí, Tarragona, Spain, 2010, 143-152.

[11] M. Montaner, B. López, J. L. de La Rosa, A taxonomy of recommender agents on the internet, *Artificial Intelligence Review* **19** (4) (2003), 285-330.

[12] J. M. Morales-del-Castillo, E. Peis, A. A. Ruiz, E. Herrera-Viedma, Recommending biomedical resources: A fuzzy linguistic approach based on semantic web, *International Journal of Intelligent Systems* **25** (12) (2010), 1143-1157.

[13] J. M. Morales-del-Castillo, R. Pedraza-Jiménez, A. A. Ruíz, E. Peis, E. Herrera-Viedma, A semantic model of selective dissemination of information for digital libraries, *Information Technology and Libraries* **28** (1) (2009), 21-30.

[14] M. Pazzani, D. Billsus, Learning and Revising User Profiles: The Identification of Interesting Web Sites, *Machine Learning* **27** (3) (1997), 313-331.

[15] P. Resnick, H. R. Varian, Recommender Systems, *Communications of the ACM* **40** (3) (1997), 56-58.

An ontology-based record linkage method for textual microdata

Sergio Martínez [a,1], Aida Valls [a] and David Sánchez [a]
[a] *Department of Computer Science and Mathematics. Universitat Rovira i Virgili*
Intelligent Technologies for Advanced Knowledge Acquisition (ITAKA) research group
Av. Països Catalans, 26. 43007. Tarragona, Catalonia (Spain)

Abstract. Disclosure control is a critical aspect when publishing information from databases (*i.e.* microdata), because they store private information about individuals. The goal of a privacy-preserving method is to avoid the re-identification of individuals from the published data. Several disclosure control methods to *mask* published data have been developed. To evaluate the quality of the anonymization process, *disclosure risk* methods measure the capacity of an intruder to link the records in the original dataset with those of in the masked one. *Record linkage* methods proposed in the literature are focused only on numerical and ordinal data. In this paper we present a new record linkage method for textual data that exploits the semantics of the values using ontologies. It relies on the theory of semantic similarity to propose linkages between the original and the masked records. The paper compares the results obtained with our method with the ones given by a traditional non-semantic approach. Evaluation shows that the semantic-based record linkage is able to better evaluate the disclosure risk of masking methods dealing with textual microdata.

Keywords. Privacy protection, disclosure risk, knowledge-based systems, ontologies, semantic similarity.

Introduction

Social and economic studies require large and detailed data (i.e., microdata) about individuals. Statistical agencies gather this data from polls, questionnaires or usage logs, which, before made public, must be properly anonymized. Assuring the protection of the identity of the individuals is a critical aspect, because, in many situations, datasets contain personal confidential information. The goal of statistical disclosure methods is to avoid that an intruder re-identifies an individual from the published data, associating or retrieving his confidential information. The anonymization methods for the protection of privacy of a person are the current means to protect the identity of that person [21]. Several anonymization techniques, based on masking original data, have been developed to minimize the re-identification risk [4].

[1] Corresponding Author.

Some masking methods have been designed, but most of them are specific for numerical data [4]. However, in many situations, some information can only be expressed by means of textual labels (e.g., which are the hobbies or the Religion of the respondent). Nowadays, with the enormous growth of the Information Society, datasets containing textual information are becoming easily available at a larger scale. On the contrary to numbers, the processing of such qualitative values cannot be done by means of arithmetic operators and require a proper semantic analysis to understand their meaning. Considering that words correspond to concepts with a concrete semantic content, knowledge sources (such as structured thesaurus, ontologies, tagged corpora, etc.) are needed to interpret concepts associated to the terms.

In recent years, some authors (see section 1) have proposed *generalization*-based masking methods dealing with textual data from a semantic point of view. In a nutshell, these works rely on semantic knowledge represented in a taxonomy, which is exploited to propose transformations of textual data, by means of the substituting specific terms by more general ones that taxonomically subsume them. For example, given a dataset in which the labels of an attribute correspond to sport names (e.g., *soccer, tennis, swimming, sailing*, etc.), these can be masked using a common label that semantically subsumes them at a certain level of generality (e.g., *team sports, water sports, sports*, etc.); in this case, a taxonomy of sports is needed to find suitable generalizations. Generalizations are done until the *k*-anonymity property is satisfied [17]. This property establishes that each record in a dataset (i.e., an individual's set of responses) must be indistinguishable with at least *k-1* other records. However, as a result of this data transformation process, an information loss occurs. *Information loss* measures the reduction of the utility of the masked data with respect to the original one [12]. So, masking methods aim generating a new anonymized version of the dataset that minimizes the information loss. In the case of textual attributes, information loss should be considered as a function of the reduction of semantic content (i.e., the more abstract the generalizations, the higher the information loss) [9,17].

There is a trade-off between the *information loss* and the *disclosure risk*. The latter measures the capacity of an intruder to obtain the information contained in the original dataset from the masked one [12]. To compute the disclosure risk, many works [4,16,20] consider *record linkage* (RL) methods. These methods try to link records in the original dataset with those in the masked one. The higher the amount of linked records, the more the risk of disclosing confidential information is. Two general record linkage methods are usually considered in the literature [12]. *Distance based record linkage* computes a distance measure between original and masked records, linking each masked record to the closest one found in the original dataset. For numerical data, an Euclidean distance is typically used [13]. *Probabilistic record linkage* [13] bases the matching on the expectation-maximization algorithm, which considers the amount of coincidences between masked and original datasets.

Classical RL methods have been defined independently to the concrete masking method used to anonymize the original dataset. However, as the disclosure risk of a masking process must be measured in the worst possible scenario, one should consider the possibility that the intruder uses especially tailored RL methods to match original and masked data. In fact, the use of generic measures underestimates the disclosure risk. Some works have shown that it is possible to increase the amount of linkages by designing especially tailored RL methods for concrete masking schemas [11,12].

Both generic and ad-hoc RL methods proposed in the literature are focused only on numerical and ordinal data. However, as stated above, several masking methods for

textual attributes have been proposed in recent years. As far as we know, no especially tailored RL methods have been proposed to evaluate the disclosure risk of masking methods dealing with textual microdata.

In this paper, we present a new distance-based RL method especially designed to measure the disclosure risk of masking generalization methods for textual attributes. Our method relies on ontologies and the theory of semantic similarity to propose linkages between the original and the masked datasets, discovering the most semantically similar records. Our RL method has been applied to evaluate the disclosure risk of a classical generalization schema, comparing it to the usual non-semantic RL implementation.

The rest of the paper is organised as follows. Section 1 reviews masking methods dealing with textual data. Section 2 describes the semantic foundations of our RL method: ontologies and semantic similarity. Section 3 presents and formalises our RL method when applied to multi-attribute textual data. Section 4 tests our approach by evaluating the disclosure risk of generalization-based masking methods. The final section contains the conclusions and several lines of future research.

1. Anonymizing textual attributes

An anonymization schema takes a dataset D consisting on m records (i.e., individuals) and n attributes (i.e. responses). These attributes can be classified in four types: *identifiers* (which unambiguously identify the individual); *quasi-identifiers* (which may identify some of the respondents, especially if they are combined with the information provided by other attributes); *confidential outcome attributes* (which contain sensitive information); and *non-confidential outcome attributes* (the rest). The goal of disclosure control methods is to prevent the link of the published confidential information to unique individuals. Before publication, identifiers (such as ID card numbers) are directly removed from the dataset. Quasi-identifiers do not link to specific respondents if they are considered separately but the problem arises if they are considered in groups (e.g., job + city of living + age). Before releasing the data, these attributes must be masked by means of an anonymization schema, resulting in a modified dataset D^A. As stated in the introduction, a classical approach is to ensure that the masked dataset is k-anonymous. The value of k defines the desired level of privacy and influences the information loss.

In the following, we review works proposing anonymization schemas focused on textual attributes. They make use of *Value Generalisation Hierarchies* (VGHs) to aid the interpretation of data and to assist the masking process [2,6,8,17]. VGHs are manually constructed taxonomical structures defined according to the input data, where the values of the attributes are leaves of the hierarchy, which are recursively subsumed by common generalisations. The masking process consists on substituting the values of the quasi-identifiers of each record by another that is more general, obtained from the hierarchical structure associated with that attribute. This generalisation process decreases the number of distinct tuples in the dataset and, therefore, increases the level of k-anonymity. The process ends when k-anonymity if fulfilled for the whole dataset.

To retain the utility of data, masking methods should select, from all the possible combinations of generalized tuples fulfilling the k-anonymity, the one that minimizes the information loss. In exhaustive approaches, the search space results in NP-hard algorithms that can only be applied to small datasets. Some authors restrict the search

space introducing different kinds of constraints [8,17]. Other authors opted by using a non-optimum heuristic search approach [8,10].

Information loss can be measured in different ways. Distributional metrics are aimed to evaluate if the data in the masked version follows a similar statistical distribution than the original values [2]. However, these metrics do not capture how semantically similar the anonymized set is with respect to the original data. Due to this reason, more recently, other authors [8,17] measured the information loss as a function of the level of generalization of each attribute in the masked dataset. The higher the generalization, the more abstract the masked dataset will be, resulting in a higher loss of semantic content. This provides more accurate assessments of the differences between the semantic content of the original and masked dataset that better retain the utility of data from a semantic point of view [9].

2. Ontology-based semantic similarity

A record linkage method, especially when tailored for a specific masking schema, can be seen as a reverse engineering process, in which an intruder tries to guess and undo the data transformation performed during the anonymization process. In the case of masking methods dealing with textual data, two parameters determine the result: the knowledge structure used to propose generalizations and the quality criteria used to guide the anonymization towards the transformations that minimize the information loss. Obviously, both elements are variables that remain hidden to the intruder. In consequence, the RL method should either try to guess them from input data or substitute them by other generic elements available.

As stated in section 1, masking methods based on value generalization exploit ad-hoc taxonomic structures constructed according to input attribute labels (VGH). As they are manually constructed, it is neither feasible nor scalable to reconstruct its structure from observed data. Instead, one can use already available knowledge structures that aim to be general enough to cover most of the concepts that may appear in a domain: ontologies. Ontologies are formal and machine-readable structures, representing a shared conceptualization of a knowledge domain, expressed by means of semantic relationships [3]. They have been successfully applied in many areas that deal with textual resources [15] and knowledge management [19].

From a domain independent point of view, one can use a general ontology like WordNet. WordNet [5] is a freely available lexical database that describes and organises more than 100,000 general English concepts, which are semantically structured in an ontological way. Concepts are linked by means of taxonomical, semantic and lexical relations. In case that input data consist on very specific terminology of a concrete domain of knowledge, one may exploit specific domain ontologies instead.

Once we have selected the knowledge source in which the RL will rely, it is necessary to define a criterion to match records between the masked and original datasets. As stated in the introduction, distance-based RL methods define a distance measure so that the closest records are matched. Ideally, this measure should be as similar as possible to the quality metric used to anonymize the data, which, as stated above, remains hidden to the intruder. In the case of generalization methods, one can assume the generalization process aims to minimize the information loss. From a semantic point of view, information loss is a function of the difference between the

degree of generality of the original and masked –generalized- values. So it can be seen as a measure of semantic alikeness (i.e., the lower the difference in the degree of generalization, the more semantically similar the original and masked values are). To define a general criterion that will guide the RL process in an inverse manner, we will rely on the theory of *semantic similarity* [7].

Semantic similarity estimates the taxonomical alikeness of terms based on semantic evidence extracted from one or several knowledge resources. Although different knowledge bases can be used, we will focus on ontologies. In ontology-based measures, ontologies are seen as a directed graph in which semantic interrelations are modelled as links between concepts, and their semantic distance can be estimated by counting the number of edges separating them. They are characterized by being easily applicable and highly efficient, lacking the constraints and dependencies on external resources that other semantic similarity paradigms present [1].

The classic way to estimate the semantic distance (i.e. the inverse to similarity) between two ontological nodes, c_1 and c_2, is to calculate the shortest Path Length connecting these elements (2) [14]. Despite its simplicity, it is a widely used measure that approximates the semantic alikeness in a very fast way. This is particularly interesting in the privacy context, where the datasets may be quite large.

$$dist_{Path\_length}(c_1, c_2) = \min \#\ of\ is-a\ edges\ from\ c_1\ to\ c_2 \quad (1)$$

3. A new record linkage method for generalization-masked datasets

In this section, we propose an ontology-based record linkage method for textual attributes named *Semantic Record Linkage* (SRL). It is especially tailored to measure the disclosure risk of masking schemas based the generalization of values, which are the most common when dealing with textual data.

Having a dataset $D = (r_1, r_2, ..., r_m)$ with records from *m* individuals, let us have that D^A is a publishable and, therefore, previously anonymized, version of *D*. We consider the typical scenario used in works like [12,18] consisting on: (i) identifier attributes in *D* have been removed in D^A; (ii) if an attribute is considered confidential then it is not modified; (iii) the anonymization is applied to quasi-identifier non-confidential attributes. Therefore $D^A = D^A_{nc} + D_c$, where D^A contains *m* records, each of them composed by *n* attributes, some of them being confidential (D_c) and the rest being non-confidential and anonymized (D^A_{nc}), so that $n = nc + c$.

Then, let us consider that a hypothetic intruder has another dataset *B* that contains *m'* records, each of them composed by *n'* attributes, one of them being identifier and the rest being non-confidential. These records have the original values of the individuals, without any anonymization. Without loss of generality, let us assume that all the records in *B* correspond to individuals that are also in *D*, the intruder can access to confidential data if he is able to correctly link a record $r_i^{'} \in B$ with the protected record $r_i^A \in D^A$, so that $r_i^{'}$ and r_i^A correspond to the same individual. This can be achieved by using the common non-confidential attributes in *B* and D^A, that is $B \cap D^A_{nc}$.

The amount of correct linkages that one can find between the anonymized and original datasets measures the disclosure risk of a privacy-preserving method.

Our *semantic record linkage* method (SRL) reproduces this scenario focused on textual attributes and value generalization anonymization methods. As stated in section 2, it relies on already available ontologies to assess the semantic distance between textual values. As a result, masked and original records can be linked calculating the closest semantic distance (or the maximum similarity) between each original and anonymized record.

Let us have that the original database *D* is composed by *m* records $r_i = (r_{i,1},...,r_{i,nc})$, and the masked version, D^A, obtained by means of a value generalization method, is composed by $r_i^A = (r_{i,1}^A,...,r_{i,nc}^A)$, where r_{ij} and r_{ij}^A are textual values. Let us have another dataset *B*, owned by the intruder, with $r_i' = (r_{i,1}',...,r_{i,nc}')$. The problem to solve is to find a record in $r_i^A \in D^A$ for each r_i' so that r_i' and r_i^A correspond to the same individual, disclosing his identity.

To evaluate the risk of identity disclosure, we act as an intruder. We search the least distant (i.e., most similar record) to r_i' in D^A, which is calculated using eq. 2. The comparison function relies on the quantification of the semantic distances between values of the non-confidential attributes. The classical *path length* function has been taken, for its computational efficiency on large ontologies. It calculates the length of the shortest *path* between pairs of textual values in a ontology (eq. 1 section 2). Notice that the solution will not be unique when several records are equally distant to r_i'.

Definition 1. The *distance between two records* r_i and r_k is defined as the arithmetic mean of the semantic distance between each of the values of their attributes:

$$record\_distance(r_i, r_k) = \frac{\sum_{j=1}^{nc} dist_{path\_length}(r_{ij}, r_{kj})}{nc} \quad (2)$$

Assuming that the anonymization has tried to preserve the semantics of original data (as explained in section 1), we apply this criterion to propose the linkages between the anonymized and the original records. Then, the disclosure risk is obtained as a function of the probability of proposing correct linkages, as follows.

Definition 2. Having $L_{r_i'} \subset D^A$ be the set of records with minimum distance with respect to the record r_i', the probability of making a correct linkage of r_i' in D^A is calculated as:

$$P_{RL}(r_i') = \begin{cases} 0 & if\ r_i \notin L_{r_i'} \\ \frac{1}{|L_{r_i'}|} & if\ r_i \in L_{r_i'} \end{cases} \quad (3)$$

being $|L_{r_i'}|$ the cardinality of the dataset $L_{r_i'}$ and r_i the original record for r_i'.

The linkage will be correct if the original record corresponding to $r_i^{'}$ belongs to the set $L_{r_i'}$. Notice that, in the case that $L_{r_i'}$ was a singleton ($|L_{r_i'}|=1$), then the linkage is unique and the individual has been re-identified ($P_{RL} = 1$).

Then, the disclosure risk of the evaluated privacy-preserving method can be measured as the difficulty in finding correct linkages between original and masked datasets. This is typically calculated as the percentage of correctly linked records [18]. In our case, it is calculated as the percentage of the average probability of linking each the *m* anonymized records, as follows (SRL):

$$SRL = \frac{\sum_{i=1}^{m} P_{RL}(r_i^{'})}{m} \cdot 100 \quad (4)$$

As illustrative example, Figure 1 shows a dataset D^A with two textual attributes considered as quasi-identifiers (*Sports* and *Job*), which have been anonymized with a level of k=2, and one numerical attribute considered confidential. Let us have that an intruder knows the information about his friend Mr. X, represented in $r_i^{'}$. He wants to link $r_i^{'}$ and D^A in order to discover the incomings of Mr. X.

Following our proposal for Semantic Record Linkage, $r_i^{'}$ is linked with the two first records in D^A. For each of them, the probability of being Mr. X is $P_{RL}(r_i') = 0.5$, which implies a significant risk of disclosure. On the contrary, if the risk was evaluated by direct matching between the values in $r_i^{'}$ and D^A, the risk of disclosure would be assumed to be 0, because there textual values are different in $r_i^{'}$ and D^A. In this latter case, we are underestimating the risk of publishing that dataset.

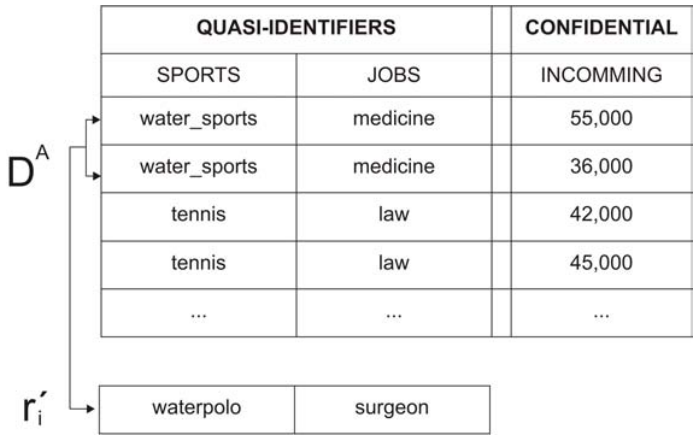


Figure 1. Example of semantic record linkage. The record $r_i^{'}$ is linked to the anonymized dataset D^A

4. Evaluation of the SRL method

In order to test the adequacy of the proposed method, and the benefits of using background knowledge in the form of ontologies to evaluate the disclosure risk of generalization methods, we have compared our SRL method against a naïve implementation (named *Matching-based Record Linkage*, MRL) in which no background knowledge is used. In this last case, the criterion to propose record linkages is based only on the matching of textual labels (i.e., the method searches for terms with identical values in both the original and masked datasets).

The dataset used for evaluation purposes consists on a set of *real* answers to polls made by the *Observatori de la Fundació d'Estudis Turístics Costa Daurada* at the *Delta de l'Ebre* Catalan National Park. The dataset comprises 975 individual records with two textual attributes. We find a total of 211 different response tuples, 118 of which were unique (i.e. identifying a single person).

To evaluate our SRL method, we have implemented a generalization method that aims to depict the methods proposed by related works (see section 1). The anonymization algorithm generates all the possible generalizations for each value and selects the one that results in a lower information loss to substitute the original value [8]. The generalization process is guided by a VGH built for the particular dataset, as done in related works.Figure 1 shows an evaluation of the disclosure risk in terms of percentage of record linkages between original and anonymized datasets. It shows the results of SRL and MRL for different anonymizations of the dataset, according to different levels of k-anonymity (with k from 2 to 20). For the SRL method, WordNet (v.2) has been used as ontology.

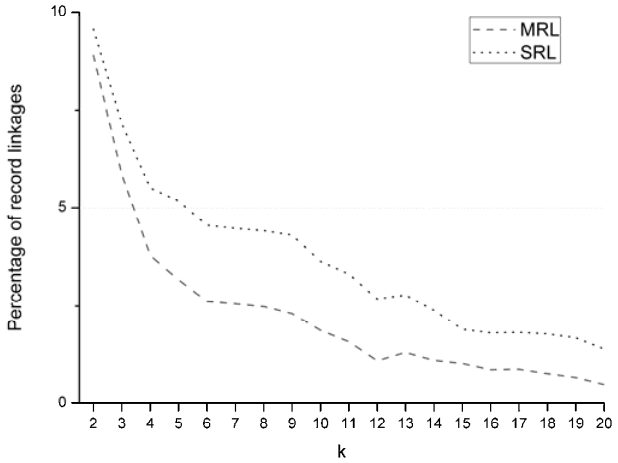


Figure 2. Percentage of correct linkages obtained with SRL and MRL methods

We can see that in this experiment the proposed SRL method is able to improve the amount of correct matchings proposed by the non-semantic approach (MRL).

Notice that, as the level of k-anonymity increases, the information loss also increases because, for example, with k=15 each record must have at least 14 other records sharing the same combination of values. For this reason, the usual level of

anonymity required to publish microdata is around 4 to 8, which preserves the privacy but maintains also a low information loss. In this interval of k, we can see that the amount of linkages obtained with SRL almost double those achieved by the MRL approach. The comparison of the results with all the k-values from 2 to 20 indicates that our SRL method better quantifies the disclosure risk of the generalization schemas dealing with textual data in all cases.

5. Conclusions and future work

In this paper we propose a novel RL approach that uses the knowledge available in ontologies to find the most feasible semantic linkages between records. Assuming that the intruder does not know the parameters used during the data anonymization process, we use publicly available general ontologies as WordNet and a widely used semantic distance as Path Length to assess the most feasible transformation performed over the masked data. The evaluation indicates that the SRL method increases the probability of record linkage. Therefore, SRL may be a useful tool to better evaluate the real disclosure risk of generalization methods dealing with textual data, in order to better guarantee the individuals' privacy protection.

The work presented in this paper opens a new line of research in RL for textual information. Knowledge representation languages and knowledge modelling paradigms provided by Artificial Intelligence enables interpreting data at a conceptual level. The semantic record linkage approach proposed in this paper is not only useful for measuring the privacy preservation in databases, but it can also be applicable to other domains that search connections between records in different databases, such as relation discovery, which do not necessarily aim at de-identifying the individuals.

From the work initiated in this paper, some points of study can be devised. Firstly, the two main components of the method should be studied in detail: (1) the knowledge base used for comparing the terms and (2) the semantic distance measure used to find the closest records. In the former, an analysis with other ontologies with different levels of granularity and different degrees of specificity could raise interesting conclusions. With regards to the second aspect of the method, a wider analysis of different semantic similarity measures available in the literature may aid to improve the results. Other path-length functions that consider more taxonomical information could be applied, such as the ones that consider the size of the ontology or the set of ancestors of the concepts compared [1]. In addition, other relatedness measures that also include non-taxonomical information could be studied. However, those features scarcely appear in the currently available ontologies.

Finally, new techniques to extend this approach not only to simple textual answers (i.e. words) but to more complex sentences will permit to use this kind of techniques in a wider range of applications, such as private information retrieval in Web Search Engines.

Acknowledgements

We would like to thank the Observatori de la Fundació d'Estudis Turístics Costa Daurada and the Delta de l'Ebre National Park (Departament de Medi Ambient i Habitatge, Generalitat de Catalunya; Department of the Environment and Housing of

the Autonomous Government of Catalonia) for providing the evaluation data. This work is supported by the Spanish Ministry of Education and Science (projects ARES – CONSOLIDER INGENIO 2010 CSD2007-00004 – and eAEGIS – TSI2007-65406-C03-02), the Spanish Ministry of Science and Innovation (DAMASK project, *Data mining algorithms with semantic knowledge*, TIN2009-11005) and the Spanish Government (PlanE, Spanish Economy and Employment Stimulation Plan). Sergio Martínez Lluís is supported by a predoctoral research grant of the Universitat Rovira i Virgili.

References

[1] M. Batet, D. Sanchez, A. Valls, An ontology-based measure to compute semantic similarity in biomedicine, *Journal of Biomedical Informatics* (2010), (in press).
[2] R. J. Bayardo, R. Agrawal. 2005. Data Privacy through Optimal k-Anonymization. the 21st International Conference on Data Engineering: IEEE Computer Society. p 217-228.
[3] P. Cimiano. *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*, Springer-Verlag New York, Inc., 2006.
[4] J. Domingo-Ferrer, A Survey of Inference Control Methods for Privacy-Preserving Data Mining, in: *Privacy-Preserving Data Mining*, C. C. Aggarwal, P. S. Yu, eds, Springer US, 2008, 53-80.
[5] C. Fellbaum. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*, The MIT Press, 1998.
[6] V. S. Iyengar. 2002. Transforming data to satisfy privacy constraints. KDD: ACM. p 279-288.
[7] J. J. Jiang, D. W. Conrath, Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy, in: *International Conference Research on Computational Linguistics (ROCLING X)*, 1997, 9008.
[8] T. Li, N. Li, Towards optimal k-anonymization, *Data & Knowledge Engineering* **65** (1) (2008), 22-39.
[9] S. Martinez, D. Sanchez, A. Valls, Ontology-Based Anonymization of Categorical Values, in: *Modeling Decisions for Artificial Intelligence*, V. Torra, Y. Narukawa, M. Daumas, eds, Springer Berlin / Heidelberg, 2010, 243-254.
[10] S. Martínez, A. Valls, D. Sánchez, Anonymizing Categorical Data with a Recoding Method Based on Semantic Similarity, in: *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Applications*, E. Hüllermeier, R. Kruse, F. Hoffmann, eds, Springer Berlin Heidelberg, 2010, 602-611.
[11] P. Medrano-Gracia, J. Pont-Tuset, J. Nin, V. Muntés-Mulero, Ordered Data Set Vectorization for Linear Regression on Data Privacy, in: *Modeling Decisions for Artificial Intelligence*, V. Torra, Y. Narukawa, Y. Yoshida, eds, Springer Berlin / Heidelberg, 2007, 361-372.
[12] J. Nin, J. Herranz, V. Torra, Rethinking rank swapping to decrease disclosure risk, *Data & Knowledge Engineering* **64** (1) (2008), 346-364.
[13] D. Pagliuca, G. Seri. 1999. Some results of individual ranking method on the system of entreprise accounts annual survey.
[14] R. Rada, H. Mili, E. Bicknell, M. Blettner, Development and application of a metric on semantic nets, *Systems, Man and Cybernetics, IEEE Transactions on* **19** (1) (1989), 17-30.
[15] D. Sánchez, D. Isern, M. Millan, Content annotation for the semantic web: an automatic web-based approach, *Knowledge and Information Systems* (2010), 1-26.
[16] F. Sebé, J. Domingo-Ferrer, J. Mateo-Sanz, V. Torra, Post-Masking Optimization of the Tradeoff between Information Loss and Disclosure Risk in Masked Microdata Sets, in: *Inference Control in Statistical Databases*, J. Domingo-Ferrer, ed, Springer Berlin / Heidelberg, 2002, 75-98.
[17] L. Sweeney, k-anonymity: a model for protecting privacy, *International Journal Uncertainty Fuzziness Knowledge-Based Systems* **10** (5) (2002), 557-570.
[18] V. Torra, J. Domingo-Ferrer, Record Linkage methods for multidatabase data mining, in: *Information Fusion in Data Mining*, V. Torra, ed, Springer, 2003.
[19] A. Valls, K. Gibert, D. Sánchez, M. Batet, Using ontologies for structuring organizational knowledge in Home Care assistance, *International Journal of Medical Informatics* **79** (5) (2010), 370-387.
[20] W. Yancey, W. Winkler, R. Creecy, Disclosure Risk Assessment in Perturbative Microdata Protection, in: *Inference Control in Statistical Databases*, J. Domingo-Ferrer, ed, Springer Berlin / Heidelberg, 2002, 49-60.
[21] R. Zallone. 2010. The privacy paradox, or how i learned to have rights that never quite seem to work. AAAI Spring Symposium Series, AAAI Publications. California, USA.

An Evaluation Framework for Location Privacy

Sergi MARTÍNEZ-BEA and Vicenç TORRA
IIIA - Institut d'Investigació en Intel·ligència Artificial
CSIC - Consejo Superior de Investigaciones Científicas
Campus de la UAB s/n, 08193 Bellaterra, Catalonia, Spain
Email: {smartinez, vtorra}@iiia.csic.es

Abstract. With the evolution and expansion of location tracking technologies such as GPS, RFID, etc. and their integration with handheld devices, lots of different applications have appeared. They use the user's location to provide more information to the application in order to commit their specific task e.g. Google Maps use user's location in order to tell him the shortest path to the her destination. However all of this location data is sensible data that could seriously compromise the user's privacy. There are different protection methods in the literature that try to reduce the risk of disclosure while reducing the loss of information as much as possible. We contribute to the existing literature with a framework for the evaluation of the protection methods, in terms of the risk of disclosure and loss of information.

Keywords. Location Privacy, Information Loss, Disclosure Risk

Introduction

In this last years, most of the new handheld devices are equipped with location tracking technologies such as GPS, RFID, GSM, etc. The expansion of those devices in the market and society is a fact. There are plenty of useful applications that use location information to improve the service they offer to the user, or applications that simply rely on location information to achieve its functionality. This location information is potentially useful for third parties, such as statistical agencies or traffic agencies. Nevertheless, this location data is sensible data as it could clearly identify the user and could let an attacker track her. Therefore, this data in the wrong hands could be potentially dangerous for the privacy of the individuals. In order to prevent the disclosure of the individual's identity, some protection methods have been developed. There are two kind of protection methods, the *perturbative* and the *non-perturbative*. The perturbative methods add noise to the data making it more difficult to re-identify the individuals from it ([1], [2], [3], and [4] are examples of perturvatibe methods). The non-perturbative methods do not modify the data, instead they use encryption systems for masking them. Our resarch is mainly focused on perturbative protection methods. Some of the existing literature such as [1], [2] and [3] only focus on protecting the data (i.e. Minimize the disclosure risk). Others, like [4], prevent the disclosure and try to preserve the data utility as high as possible (i.e. Minimize the information loss). There are also other approaches such as [9], [10],

and [11]. The disclosure risk is a metric used to measure the risk of an attacker that has acquired the protected data, to re-identify the idividuals. The information loss is a metric used to measure the loss of information caused by the perturbative protection methods. Evaluating the protection methods is very important as it allows you to compare them and get a value to determine the level of protection it can achieve as well as the amount of information that is still in the data. On most of the current literature, the protection methods are evaluated with different disclosure risk and information loss measures, depending on the author. Specifically, the information loss measures are highly related to the use that the author thinks the data will be given, which is not the best option because they are restricting a lot the utility of the data and, moreover, we can not always know which use the data will be given. In this work, we present a full evaluation framework that provides measures that are intended to be as generic as possible.

This paper is structured as follows. In section 1 we present the evaluation measures for disclosure risk and information loss. Then, in section 2 we detail the framework structure and we give an explanation for each relevant class. In section 3, we provide a *Proof of concept* of the framework, by testing it with a given dataset. Finally, to conclude this document, we talk about the conclusions and further work.

1. Evaluation measures

As stated before, having evaluation measures for the protection methods is important, as it allows you to determine the protection level achieved and the amount of information that has been lost in the process. It is also important that if we do not know the use that the data will be given, the evaluation measures should be as much generic as possible. They should reflect the data utility but without being too specific. Otherwise, the evaluation results are biased towards those specific uses. In this framework, we provide generic information loss measures as well as a disclosure risk measure, both presented in [5] and defined below.

In order to understand better the formulas presented below, we first give the formal definition of a trajectory. A trajectory can be defined as a sequence of places or spots that a given object or person has gone through. We will define each place or spot, as a point, with their coordinates x and y and a timestamp t that determines the time when the object or person was there. Therefore, we can define formally a trajectory as:

$$\{(x_1, y_1, t_1), (x_2, y_2, t_2), \ldots, (x_n, y_n, t_n)\} \text{ where } n \in \mathbb{N}$$

1.1. Information Loss

In order to define a generic measure, we define it in terms of three partial measures, IL_1, IL_2, and IL_3 focusing on three different aspects. Those partial measures are defined as:

- IL_1. It is composed by the measures $IL_{1.1}$ and $IL_{1.2}$, which are defined below, and it is defined as:

$$IL_1 = \frac{IL_{1.1} + IL_{1.2}}{2} \tag{1}$$

* $IL_{1.1}$ is the average of the difference between the means of both original and protected trajectories and is defined as follows:

$$IL_{1.1} = \frac{1}{2s}\left(\sum_{i=1}^{s} \frac{|\mu_{x_i} - \mu'_{x_i}|}{Max(|\mu_{x_i}|, |\mu'_{x_i}|)} + \frac{|\mu_{y_i} - \mu'_{y_i}|}{Max(|\mu_{y_i}|, |\mu'_{y_i}|)}\right) \tag{2}$$

where s is the number of trajectories, μ_x the average of the X coordinate, and μ_y the average of the Y coordinate.

* $IL_{1.2}$ is the average of the difference between the autocorrelation functions [5] of both original and protected trajectories. That is,

$$IL_{1.2} = \frac{1}{4} \sum_{h=0,n/4,n/2,3n/4} \left(\frac{1}{s}\sum_{i=1}^{s} \frac{|\rho_i(h) - \rho_i(h)'|}{Max(|\rho_i(h)|, |\rho_i(h)'|)}\right) \tag{3}$$

where s, μ_x, μ_y are the same as before, and the $\rho_i(h)$ is the *Adapted Autocorrelation Function* [5].

- IL_2 is defined as the absolute differences between the original and protected data. That is,

$$IL_2 = \frac{1}{2 \times s \times n} \sum_{i=1}^{s \times n} \left(\frac{||x_i| - |x'_i||}{Max(|x_i|, |x'_i|)} + \frac{||y_i| - |y'_i||}{Max(|y_i|, |y'_i|)}\right) \tag{4}$$

where s is the number of trajectories, n the number of points of a trajectory (i.e. its length), x and y the coordinates of a point that belongs to a non protected trajectory and x' and y' the coordinates of a point that belongs to a protected trajectory. Each pair of x, y - x', y' has the same timestamp t.

- IL_3 is the *Simplified Space Distortion* measure (SSD) [5]. As the name indicates, it is a simplified version of the *Space Distortion* measure [4]. The SSD is defined as

$$SSD = \sum_{i=1}^{s} \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2}$$

where s is the number of trajectories, x and y the coordinates of a point that belongs to a non protected trajectory and x' and y' the coordinates of a point that belongs to a protected trajectory. Each pair of x, y - x', y' has the same timestamp t.

1.2. Disclosure risk

To evaluate the protection level achieved by the protection methods we use the disclosure risk measure. This measure gives you a value that represents the risk of re-identification between the original data and the protected data. In our case, it is computed with the *Record Linkage* method [6]. The *Record Linkage* method tries to link the protected data to the original data by computing the distance between all the pairs, using a *one-against-all* approach, and taking the pair with the minimum distance between them, as a correct match. The *Record Linkage* method for trajectories is detailed in Algorithm 1.

Algorithm 1: Trajectory Record Linkage

Input: X: original data set, X': protected data set
Output: dr: percentage of well linked pairs, LP: linked pairs
1: **foreach** $a \in X$ **do**
2: $b' = arg\_min_{b \in X'} d_{tr}(a, b)$
3: $LP = LP \cup (a, b')$
4: **end foreach**
5: $dr = \frac{\#LP}{\#X} * 100$

The values returned by the algorithm represent the percentage of correct matches that have been obtained after applying it against the original and proteced data, and the list of linked pairs.

2. Framework structure

In this work, it is important to provide some details of its implementation, to understand its strengths. The framework is designed to be multiplatform and flexible. It is implemented in *Java* so it can be executed in any platform as long as it runs the *Java Virtual Machine*. Regarding the flexibility, the important classes for the evaluation framework are implemented as *Factory Method Patterns* [7], so it has several classes that can be replaced by other custom classes without having to change anything from the rest of the framework. The relevant classes of the framework are described below. Also, In Figure 1, the simplified UML class diagram that shows the structure of the framework can be found.

The framework returns the values of the information loss and disclosure risk measures, as well as a *score* that is defined as the average of them.

2.1. Dataset Manager

The job of this class, as the name indicates, is to manage the datasets that will be used to make the desired experiments. It provides two methods, one for loading the original data, and another to save the protected data. You can replace this class so you can use any format for the dataset you need. The framework provide a very concrete data structure to load the datasets into, so it is compatible with the rest of the classes. This class is provided by default with the framework.

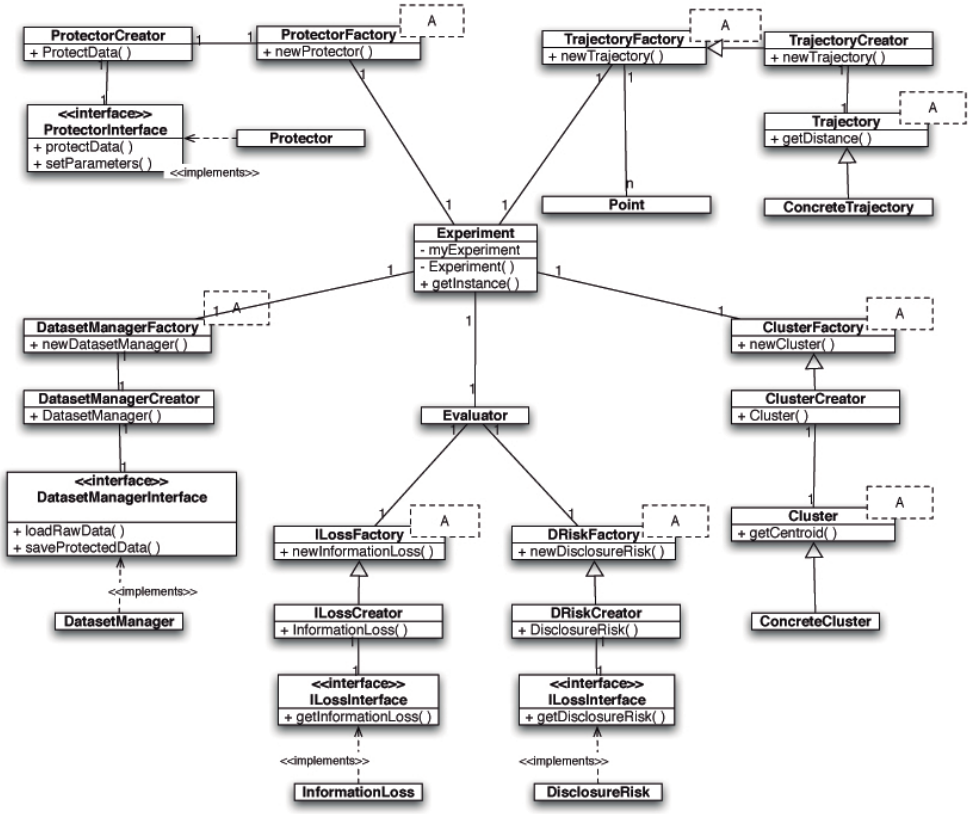


Figure 1. Simplified UML class diagram of the framework.

2.2. Cluster

This class has two main functions. One is to represent the data structure that defines groups of trajectories. The other one, is to provide the functionality that clusters have, i.e. the computation of the centroid. As the computation of the centroid of a cluster will depend on the clustering algorithm used, this class can also be replaced, so the method used to compute the centroid can be customized. This class is provided as default with the framework.

2.3. Information Loss

As the name suggests, the job of this class is to compute the information loss measure. It provides the implementation of the measures detailed on section 1.1. To provide flexibility, it is possible to replace it with another class so that the user can add more components to the measure, or use a completely different measure. This class is provided by default with the framework.

2.4. Disclosure Risk

The objective of this class is to compute the disclosure risk measure. It provides the implementation of the *Record Linkage* algorithm for trajectories, wich is described in Algorithm 1. It can also be replaced so you can add, modify, or replace the disclosure risk measure by the another desired method. This class is provided by default with the framework.

2.5. Protect Data

This class implements the protection method. It has two methods, the protection method itself and another one to set the parameters of the protection method. This class, as the others described in this section, can be replaced, so you can provide any desired protection method in order to evaluate it with the framework. This class is the only one that is not provided with the framework, as it represents the protection method to be evaluated, so it is mandatory to include a class that implements the protection method. Despite that, an example is provided with the framework. [1]

3. Experiments

In this section we provide a *Proof of concept* of the framework. We test the framework with two different protection methods. The first protection method is the one presented in [5], and the second is the one presented in [4]. The dataset used contains 43 synthetic but realistic trajectories that have been generated with the Brinkhoff generator [8].

In the results shown below we call each method with the name of its first author, so S. Martinez-Bea is the method presented in [5], and J. Domingo-Ferrer the method presented in [8].

Both protection methods have parameters that let you tune its performance. We used the *Adapted Short Time Series Distance* and the *(X median, Y)* for S. Martinez-Bea's method, and a value of 50 for the *time threshold* and *space threshold* for the J. Domingo-Ferrer's method.

In Figure 2 we can observe the Information Loss evaluation for both protection methods. The X axis represent the k values which are cluster size, and the Y axis the value of the Information Loss expressed as a precentage. It can be appreciated that in S. Martinez-Bea's method the Information Loss increases as the k value increases too, while in J. Domingo-Ferrer the Information Loss seems constant, but it presents subtile changes as it can be observed in table 2. In general S. Martinez-Bea's method performs better than J.Domingo-Ferrer's method regarding the Information Loss.

Having a look at Figure 3, we can observe the Disclosure Risk evaluation for both protection methods. The X axis represent the k values, which are the cluster size, and the Y axis the value of the Disclosure Risk expressed as a percentage. It can be appreciated that in S. Martinez-Bea's method the Disclosure Risk decreases as the k values increase, wile in J. Domingo-Ferrer the Disclosure Risk keeps constant for all k values. In this case, the J. Domingo-Ferrer's method performs better than S. Martinez-Bea's method.

[1] The framework will be soon available for its use at www.ppdm.cat

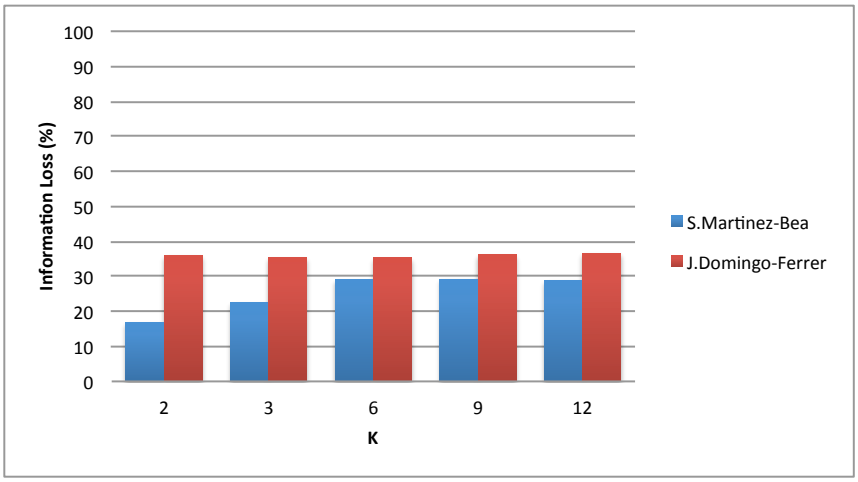


Figure 2. Results of the Information Loss evaluation.

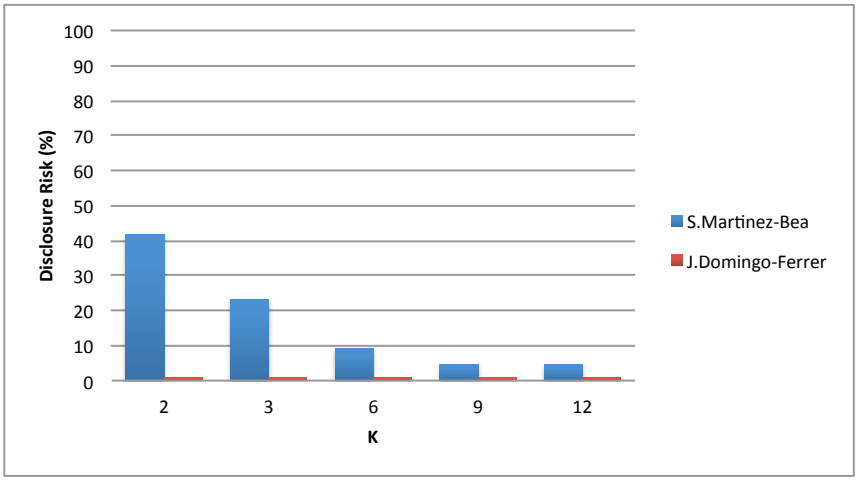


Figure 3. Results of the Disclosure Risk evaluation.

Now, going to Figure 4 we can appreciate the Score evaluation. The Score is calculated by doing the arithmetic mean between the Information Loss and Disclosure Risk measures. As Information Loss and Disclosure risk are opposite measures (i.e. If the Information Loss increases, the Disclosure Risk decreases, and viceversa), the maximum Score value that can be achieved is 50. It is important to keep in mind that in the Score, as well as in the Information Loss and Disclosure Risk, the lower value a method gets the better it is. In this Figure we can observe a comparison between the two methods, which is one of the main uses of the framework. J. Domingo-Ferrer's method is more or less constant for all k values, while S. Martinez-Bea's method have larger Score values for low k values and, as k increases, it gets a better performance. For $k = 2$, $k = 3$,

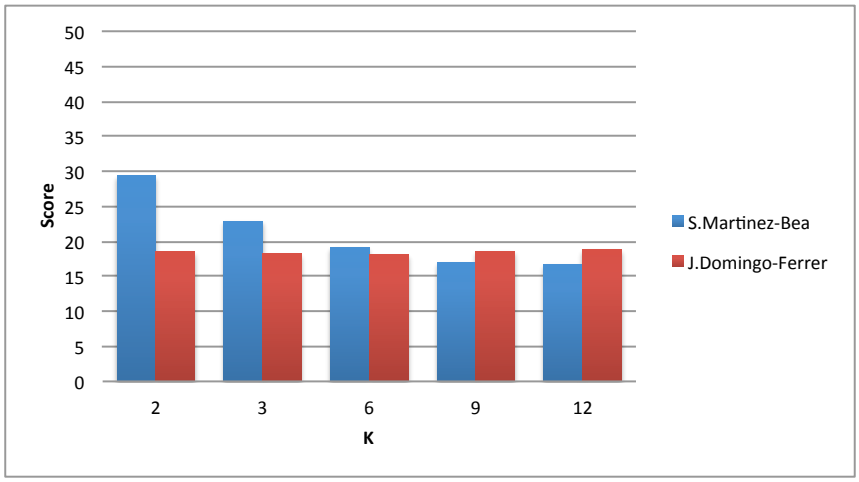


Figure 4. Results of the Score.

and $k = 6$ J. Domingo-Ferrer's method gets a better performance than S. Martinez-Bea's method, while for $k = 9$ and $k = 12$ S. Martinez-Bea's method gets a better performance than J. Domingo-Ferrer's method.

Tables 1 and 2 present the numeric results of the evaluation.

Table 1. Values for S. Martinez-Bea's protection method.

k	**Information Loss**	**Disclosure Risk**	**Score**
2	16,92	41,86	29,39
3	22,52	23,26	22,89
6	28,98	9,3	19,14
9	29,18	4,65	16,92
12	28,72	4,65	16,68

Table 2. Values for J. Domingo-Ferrer's protection method.

k	**Information Loss**	**Disclosure Risk**	**Score**
2	36,03	0,95	18,49
3	35,46	0,95	18,21
6	35,31	0,95	18,13
9	36,26	0,95	18,60
12	36,67	0,95	18,81

4. Conclusions and future work

In this document, we presented an evaluation framework for trajectories, that allow you to evaluate the protection methods, by providing a measure for the risk of disclosure of the protected data and a measure for the loss of information caused by the perturbative

method under evaluation. This framework is suitable for evaluating perturbative protection methods no matter the use that will be given to the protected data. The *Javadoc* documentation is under development and, as future work, new upcoming measures for the information loss or disclosure risk computation can be included in order to make the framework even more generic. Another improvement is to include a graphical user interface to represent both original and protected trajectories.

Acknowledgements

Partial support by the Spanish MICINN (projects eAEGIS TSI2007-65406-C03-02 and ARES - CONSOLIDER INGENIO 2010 CSD2007-00004) is acknowledged.

References

[1] G. Andrienko, N.Andrienko, F. Giannotti, A.Monreale, and D. Pedreschi: Movement data anonymity through generalization, *Proceedings of the 2nd SIGSPATIAL ACM GIS 2009 International Workshop on Security and Privacy in GIS and LBS.* ser. SPRINGL (2009), p. 27–31.

[2] G. Ghinita: Private Queries and trajectory anonymization: a dual perspective on location privacy, *Transactions on Data Privacy* vol. 2 (April 2009), p. 3–19.

[3] D. Lin, E. Bertino, R. Cheng, and S. Prabhakar: Location privacy in moving-objects environments, *Transactions on Data Privacy* vol. 2 (April 2009), p. 12–46.

[4] J. Domingo-Ferrer, M. Sramka, and R. Trujillo-Rasúa: Privacy-preserving publication of trajectories using microaggregation, *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS* ser. SPRINGL (2010).

[5] S. Martínez-Bea, V. Torra: Trajectory anonymization from a time series perspective, *Proceedings of the IEEE International Conference on Fuzzy Systems 2011, in press*.

[6] W. E. Winkler: Re-identification methods for masked microdata, *Springer* vol. 3050 (2004), p. 216–130.

[7] E. Gamma, R. Helm, R. Johnson, and J. Vlissides: Design Patterns: Elements of Reusable Object-Oriented Software, *Addison-Wesley Professional Computing Series* (1994) p. 107.

[8] T. Brinkhoff: Generating traffic data, *Bulletin of the technical comittee on Data Engineering, IEEE Computer Society*, no. 26 (2003).

[9] B. Hoh: Protecting location privacy through path confusion, *SECURECOMM âĂŹ05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, p. 194-205 (2005).

[10] B. Gedik and L. Liu: Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms, *IEEE Trans. Mob. Comput.*, p. 1-18 (2008).

[11] M. E. Nergiz and M. Atzori and Y. Saygin, Towards Trajectory Anonymization: a Generalization-Based Approach, *Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS*, p. 52-61, (2008).

A MAJOR DEPRESSION PATIENT EVOLUTION MODEL BASED ON QUALITATIVE REASONING

Francisco MUGICA [a], Solmaz BAGHERPOUR [a], Àngela NEBOT [a,1],
Antoni SERRANO-BLANCO[b,c] and Luisa BALADÓN[b]
[a]*Dept. Llenguatges i Sistemes Informàtics, Soft Computing Group, Universitat Politècnica de Catalunya, Barcelona, Spain*
[b]*Parc Sanitari Sant Joan de Déu, Sant Boi de Llobregat, Spain*
[c]*Red de Investigación en Actividades Preventivas y Promoción de la Salud, Spain*

Abstract. Providing monitoring and support to patients suffering from Major Depression plays a significant role in preventing reoccurrence and relapse of this disease which are very common characteristic of it. In this work the conceptual design of a Major Depression Remote Intelligent Monitor, called MADRIM is presented. The main goal of MADRIM is to follow the evolution of the patient during his/her recovery in order to understand its behavior and to support preventing the reoccurrence of depression. In this paper one of MADRIM's main modules, i.e. the Patient Evolution Model (PEM), is described in detail. The PEM, based on qualitative reasoning, studies the tendency in the depression level change of each patient. MADRIM is based on different input sources such are clinical data, life events and patient's mood as well as physiological data collected from sensors. As output the system provides three different levels of patient's enhancement information, i.e. progress information, alerts and alarms, to the different actors involved in the treatment, i.e. patients, primary care physicians, psychiatrists and virtual assistants.

Keywords. Major depression, qualitative reasoning, remote intelligent monitoring, MADEP

Introduction

Major depressive episode is a common mental disorder that presents with depressed mood, loss of interest or pleasure, feelings of guilt or low self-worth, disturbed sleep and/or appetite, low energy, and poor concentration. These problems can become chronic or recurrent and lead to substantial impairments in an individual's ability to take care of his or her everyday responsibilities. At its worst, depression can lead to suicide, a tragic fatality associated with the loss of about 850.000 lives every year [1]. The risk of the recurrence of major depressive disorder progressively increases with each successive episode and decreases as the duration of recovery increases.

One of the major risk factors for repeat episodes (either relapse or recurrence) is the presence of residual symptoms that persist after an episode ends; these residual symptoms tend to progress over time to another full-blown depressive episode [2, 3]. Depressive relapse is defined as an episode of major depressive disorder that occurs within 6 months after either response or remission, while recurrence is defined as another depressive episode that occurs after 6 months have elapsed.

[1] Àngela Nebot: Universitat Politècnica de Catalunya, Edifici omega, Despatx 324, c/ Jordi Girona 1-3, Barcelona 08034. E-mail: angela@lsi.upc.edu; http://www.lsi.upc.edu/dept/investigacion/sectia/soco/index.php/

Considering mentioned facts early detection of possible relapses becomes difficult after patients recovery, due to lack of enough supervision on patients status after recovery. Therefore monitoring patients remotely in their homes during the first year and during all stages of treatment gains a high importance.

Telepsychiatry, the use of telecommunications to provide psychiatric health information and care across distance, is an increasing method of providing remote medical and psychiatric care nowadays. Advances in information and communication systems, sensors and network technologies and intelligent systems are contributing to so many fields including telepsychiatry. The use of telepsychiatry, together with intelligent monitoring systems, can play a significant role in providing continuous remote monitoring during depression treatment at patients' homes and it can contribute to early detection of patient's changes. Furthermore, after a while the monitor will have available a rich dataset to be studied further in order to find out more detailed improvement patterns of patients.

In telemedicine since patients are mainly dealing with physiological symptoms, diagnosis is encountered with capturing vital physiological signals in an invasive way [4]. Some studies in telepsychiatry and depressed mood are also working on developing psychological factors based on physiological data [5-9]. However, they are still in the research steps and they are very costly for implementing at home settings in large scale or they have been designed for hospital setting. Depression is a multi-factorial condition, having numerous psychological symptoms that may or may not be present in each case. It also has multiple degrees to which a symptom is present and multiple degrees in the severity of the disorder as a whole. Another characteristic of it is that due to subjectivity of its diagnosis tools and variation of effect of antidepressant treatment on different people makes it difficult to define general improvement paths and, all these characteristics make it both complex and fuzzy. Its main practical diagnosis tool is self-report questionnaires and oral consultation and its diagnosis and symptoms are highly uncertain in nature.

Some recent studies in this area are focusing on the importance of psychological aspects to assess changes in depression severity but as secondary depressive symptoms coexisting with other chronic disease, so it is mainly focusing on a portion of symptoms and factors of depression [10]. Current studies are either focusing on physiological symptoms or they are designed for depression in elder people or depression as secondary disease. None of these studies are focusing on providing practical and less costly home monitoring systems taking into account both, psychological and physiological factors in order to decrease the chance of reoccurrence through monitoring patient's changes, finding improvement patterns and providing the support patients might need. Therefore, there is a need to develop a specific model (capable of dealing with uncertainty) for monitoring depression at home, based on both physiological and psychological factors, which allows to understand the evolution of the patient during his/her recovery and to support preventing the relapse of depression. Currently we are working in the design and development of an intelligent monitor with these characteristics in the context of the MADEP project [11]. MADEP is a Catalan research project whose goal is the design and development of a computational tool to support remote treatment and help people with depression. The main characteristics of this tool are described in [12].

It is possible to find in the literature some research that use soft computing techniques to deal in a way or another with major depression problems. In [10] a model for multiple assessments of depressive symptoms which are common also to

obstructive sleep apnea (OSA) is developed. They design a multi factor fuzzy logic model to assess and predict depression in patients with OSA. The input data of their model is based on a questionnaire that they designed for only common symptoms between depression and OSA. However, their model is not designed specifically for depression and does not use physiological data from sensors. Other studies use fuzzy logic systems to recognize activities in a home environment by using a set of sensors. The goal is to provide pervasive home monitoring for elder people [13].

Another interesting study uses support vector machine classifiers to detect depression from facial actions and vocal prosody [6]. They believe that behavioral observations are strong indicators of psychological disorders and compared clinical diagnosis of major depression with automatically measured facial actions and vocal prosody in patients undergoing treatment for depression. Their results support the idea of using automatic diagnosis clinical depression systems.

In this paper the conceptual design of a major depression remote intelligent monitor is outlined in section 1 and one of its main modules, i.e. the Patient Evolution Model (PEM), is described in detail in section 2.

1. MADRIM: MAjor Depression Remote Intelligent Monitor

As has been already mentioned before, the idea behind MADRIM is to develop a multi-factor monitor that allows following the evolution of the patient during his/her recovery in order to understand its behavior and to support preventing the relapse of depression. This will be done using the most available technology in order to decrease the costs and become a practical and useful tool for patients and physicians. MADRIM general structure design is presented in Fig. 1.

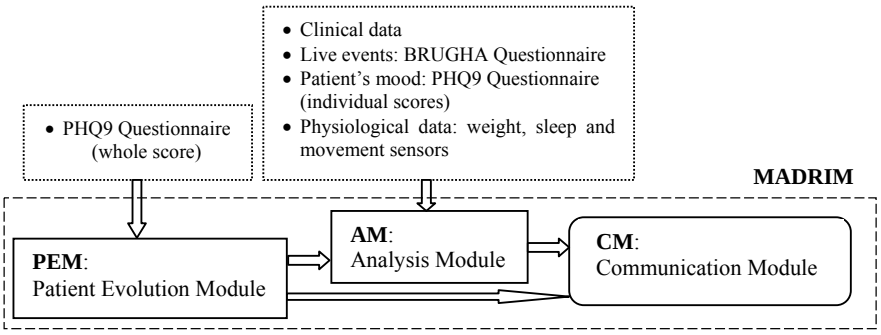


Figure 1. MADRIM general structure.

MADRIM receives the following information (input data): the clinical data of the patient (personal information and clinical history); patient's mood and live events that affect the mood obtained from patient's responses to specific questionnaires (PHQ-9 and Brugha) and physiological data steaming from weight, sleep and movement sensors.

The PHQ-9 is a depression assessment tool, which scores each of the 9 diagnostic criteria of mental disorders. The questionnaire is designed to assess the patient's mood over the last 2 weeks [14]. It is important to notice that the PHQ-9 is an extensively used questionnaire that has been validated in several studies and that is useful not only for major depression diagnosis but also for patient's evaluation. It has also been proved that this kind of questionnaires can be applied also through the telephone obtaining reliable results [15].

The Brugha questionnaire is a self-report questionnaire that examines the incidence of 12 categories of negative life events over the previous 6 months [16]. The questionnaire assesses life stressors involving moderate or long-term threat such as illness or injury, death of a close friend or relative, unemployment, financial loss and loss of important relationships. The use of this questionnaire in our system is not to support the initial diagnosis, but to help understand certain variations in patient evolution that would be inexplicable without this information.

MADRIM processes these inputs by using several modules and provides the information required by the own patients, primary care physicians, psychiatrists and other possible agents as virtual assistants, for supervising patient's treatment. The MADRIM modules are: the patient evolution module (PEM), based on a qualitative reasoning model; the analysis module (AM), based on expert knowledge and pattern recognition models and the communication module (CM), as shown in Fig. 1.

The PEM allows to trace the progress of the patients in a short time basis (1 month) and to characterize their re-establishment to the mental health pattern. PEM is centered on the overall rate of PHQ-9 questionnaire and is explained in more detail in section 3.

The AM receives the short-term patient evolution status, which is the output of the PEM. From this knowledge and together with the input data described before (doted square in Fig. 1), it defines a framework for assessing both the process of healing and the patient's risk level for each stage of treatment. The word risk is used here in the sense of defining the level of patient's enhancement, so it has nothing to do with the risk of stop taking the pills. The AM is a rule base model that uses the psychiatrist expert knowledge and patient's recovery patterns obtained from the patient evolution model. An example of AM rule available in any of the treatment stages could be the following one: **If** the patient improves normally (PEM output) **and** non-negative lives events occurred (Brugha=0) **and** non-suicide events in the past (clinical history) **then** risk is low (green light).

Finally, the CM is the responsible of organizing the risk data derived from the AM and provides the adequate information to the different actors involved in the treatment process, i.e. patient, primary care physicians, psychiatrists and virtual assistants. Fig. 2 describes this idea in a schematic way.

In order to facilitate the patient's evolution understanding and primary care physician work, a colour code representation is used by the communication module. A green light means that the patient is progressing adequately. A yellow light represents an alert.

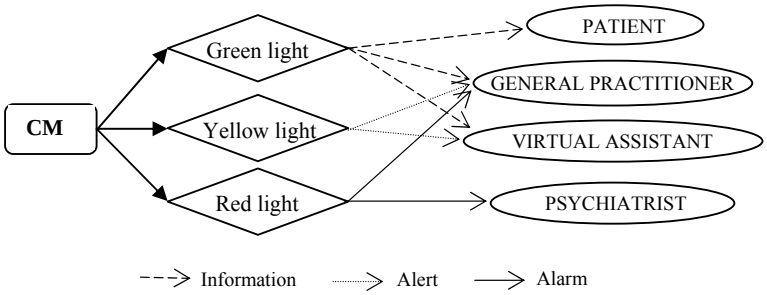


Figure 2. Output of the Communication Module.

This means that the patient is doing well but that there is the possibility that in the near future the progress of the patient suffers a recess. Therefore, the patient should be closely observed by the virtual assistant and/or the primary care physician. An example of yellow light could be when the PEM gives as output that the patient is improving quickly but a terrible live event, captured by the Brugha questionnaire, has occurred recently. In this case the AM conclude that a yellow light is the adequate risk level and sends this information together with the reasoning performed by the model to the CM.

Finally a red light represents an alarm. An alarm means that the patient is not doing well and that physicians need to take actions, i.e. have a personal interview with the patient to asses him/her more deeply, change de medication, increase the dosage, etc.

The information derived from a green light (uncontinuous arrows in Fig. 2), is presented to the patient as encouragement to continue the treatment and as progress information to the primary care physician and the virtual assistant. The last one can use this knowledge at its own convenience.

The information related to the yellow light (doted arrows in Fig. 2), is sent to the primary care physician and the virtual assistant, which decide what to do in order to perform a more accurate surveillance of the patient. It is possible that the primary care physician decides to schedule a new patient's visit in the near future and that the virtual assistant decides to ask more questions about patient's mood and give them some kind of support.

Finally, the red light information (continuous arrows in Fig. 2) is sent to primary care physicians and psychiatrists, specifying the reasoning underlying the red light.

2. PEM: Patient Evolution Module

The goal of the patient evolution model is to capture the patient's evolution patterns during treatment. Psychiatrists do not know the patient's recovery patterns from a short time perspective. It is well known that a high percentage of patients recover fully from a major depression episode after three months of medication, but it is not known the patterns of improvement that the patients follow during this period.

However, this knowledge is highly relevant if the goal is to develop a monitoring system that follows patient's progress in a daily basis. Moreover, it is undesirable to

wait three months in order to know if the patient is doing well and take actions, like change medication, if it is not the case. Precisely, the usefulness of a major depression remote intelligent monitor is to give constant feedback of patient's evolution to all the actors involved in the treatment process.

Therefore, it becomes necessary to develop a model capable of capturing all the possible patient's evolution patterns. This research deals with this challenge taking inspiration from qualitative reasoning. PEM is exclusively based on the whole score of the PHQ-9 questionnaire. A PEM based in each individual question of the PHQ-9 does not make sense at this point, because the total PHQ-9 is an instrument for measuring the depression level. So it is clinically valid as a whole.

A set of three PHQ-9 measures is used, that correspond to a period of one month, due to the fact that the patient responds the PHQ-9 questionnaire every two weeks. This idea is shown in Fig. 3.

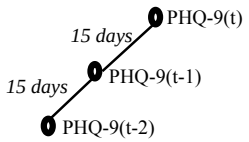


Figure 3. Period used by the PEM to evaluate short term evolution of the patient. Three PHQ-9 measures at present time (t), fifteen days ago (t-1) and one month ago (t-2) are taken into account in the model.

The period of one month, including three different time measures of the PHQ-9 questionnaire are chosen in order to omit normal fluctuations of mood. But at the same time the qualitative model is designed in a way that omitting normal fluctuations doesn't lead to losing important information in providing alarms. Three consecutive questionnaires measures are chosen for assessing changes instead of two, because two points are not enough neither for omitting normal fluctuations nor for deciding about changes.

Three main variables are defining each and every state of changes in patient's status. These three variables are direction, velocity and quality of the changes. The direction describes the tendency of the patient's changes in the PHQ-9 measures and is discretized into three classes, i.e. improvement, worsening and without changes (stuck). The velocity describes the speed of the patient's PHQ-9 changes and is also discretized into three classes, i.e. quick, slow and normal. Finally the quality of change describes the shape and can have three possible representations, i.e. oscillated, maintained and intuited. Fig. 4 describes the meaning of the three classes of both direction and quality of change variables.

The velocity is defined by means of three constants that represent the PHQ-9 change at time t and t-2. If we define Δ= PHQ-9(t-2) - PHQ-9(t), then the slow class is characterized by the equation $k1 < |\Delta| \leq k2$; the normal class by $k2 < |\Delta| \leq k3$ and the quick class is defined as: $|\Delta| > k3$.

k1 is the significant improvement value in terms of PHQ-9 which is defined based on experts knowledge, i.e. psychiatrists. k2 and k3 are the amounts of change in the depression level that correspond to a normal and fast change borderlines from a psychiatric perspective.

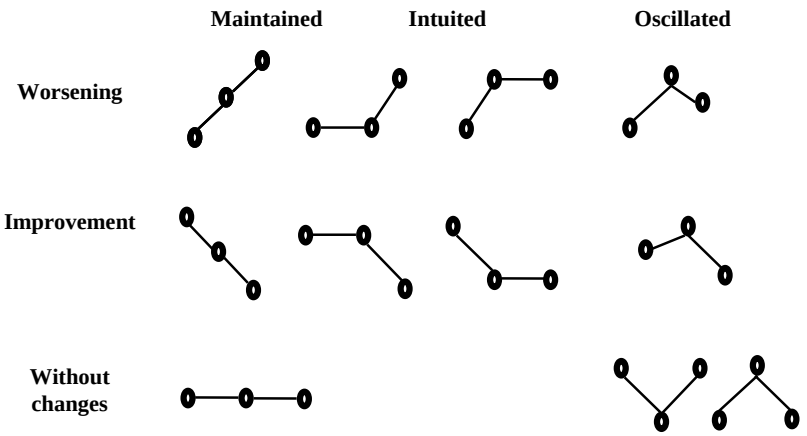


Figure 4. Output of the Communication Module.

Table 1 presents an example of two rules that are part of the patient evolution model. The terms Δ_1 and Δ_2 are defined as Δ_1= PHQ-9(t-1) - PHQ-9(t) and Δ_2= PHQ-9(t-2) - PHQ-9(t-1).

Table 1. An example of two rules that compose the PEM

Direction	Velocity	Quality
Improvement $\Delta > k1$	Quick $\|\Delta\| > k3$	Oscillated $\Delta_1 \cdot \Delta_2 < 0$
Without changes $-k1 \leq \Delta \leq k1$	Not applicable	Maintained $\Delta_1.\Delta_2>0 \vee (\|\Delta_1\| \leq k1 \wedge \|\Delta_2\| \leq k1)$

The first rule defines a patient that is "*improving quickly in an oscillation manner*". The second one corresponds to a patient that has a behavior of "*no changes in a maintained form*".

The different evolution patterns shown in Fig. 4 combined with the velocity variable compose the short term patient evolution model that is composed of 20 rules as the ones shown in Table 1. Therefore, the output of the PEM is a description of the progress of the patient in terms of the values of each of the previous variables, i.e.

direction, velocity and quality of change. An important feature of this module is its ability of self-adjust the interpretation of changes during the different stages of the treatment.

There are four treatment stages: initial, partial response, response and remission, which are defined by patient's evolution behavior. The initial stage is the period of time while there is no evidence that the medication is acting. In the partial response stage the antidepressant starts working but a reduction of 50% of the PHQ-9 initial patient's value is not reached yet. A reduction of 50% allows assuring clinically that the medication is acting. The response stage starts when the 50% of the PHQ-9 initial patient's value reduction is achieved. Finally, the remission phase starts when the patient is fully cured and should continue taking the antidepressant for a while (6 months to 1 year). PEM is able to characterize these stages using the patient evolution rules described before interpreting the rules in each phase. For example, the second rule described in Table 1 has different interpretation if occurs in patient's initial or response stages. If the patient is in the initial stage a *"no changes in a maintained form"* behavior is reasonable and a green light is set. However, this is not a desirable behavior when the patient is in the response stage. In this case a yellow light is chosen to alert primary care physician and the virtual assistant of this fact. Therefore, conceptually the PEM is composed of 80 rules (20 for each stage). The PEM has been validated by psychiatrists of the Parc Sanitari Sant Joan de Déu and it is expected that it will be validated with real patients in the near future. Each rule that conform the PEM has been discussed and evaluated with the psychiatrists and each constant value has been defined following their advice. As soon as the whole prototype is applied to specific patients it would be possible to perform a more precise and correct validation. We think that this will happen in the next months.

3. Conclusions

In this research a major depression remote intelligent monitor, called MADRIM is presented in its design phase. MADRIM is composed of three modules: the patient evolution module (PEM), the analysis module and the communication module. The PEM has been already developed and is described in detail in this paper. The main goal of this module is capture the patient's evolution patterns during treatment in a short time perspective. This is a novel approach, since it is well known that a high percentage of patients recover fully from a major depression episode after three months of medication, but it is not known the patterns of improvement that the patients follow during this period. Therefore, the characterization of short term patient's evolution patterns is crucial in order to develop a remote monitor capable of following the evolution of the patient during his/her recovery with the purpose of understanding its behavior and support preventing the relapse of depression.

The PEM has been validated by the psychiatrist of the Parc Sanitari Sant Joan de Déu and in the following months it is schedule to validate it by means of real patients of the same hospital.

The analysis and the communication modules are also discussed in this paper. The analysis module receives the short-term patient evolution status, which is the output of the PEM, and together with other input data (clinical history, PHQ-9 and Brugha questionnaires and physiological data), defines a framework for assessing both the process of healing and the patient's risk level for each stage of treatment. The communication module is the responsible of organizing the risk data derived from the analysis module and provides the adequate information to the different actors involved in the treatment process, i.e. patients, primary care physicians, psychiatrists and virtual assistants. To this end, a colour code representation is used, where a green light represents that the patient is progressing adequately, a yellow light represents an alert and a red light an alarm. MADRIM also provides the information related to the underling reasoning. The next steps are the development of the analysis and communication modules and the application of the MADRIM prototype first to simulated patients and afterwards to real patients.

Acknowledgments

This research has been supported by the Generalitat de Catalunya under project RD09-1-0013 and MICINN under project TIN2009-13895-C02-01.

References

[1] http://www.who.int/mental_health/management/depression/definition/en/

[2] A.A. Nierenberg, T.J. Petersen and J.E. Alpert, Prevention of Relapse and Recurrence in Depression: The Role of Long-Term Pharmacotherapy and Psychotherapy, *J Clin Psychiatry* **64** (2003), 13-7.

[3] D.A. Solomon, M.B. Keller, A.C. Leon, T.I. Mueller, P.W. Lavori, M.T. Shea, W. Coryell, M. Warshaw, C. Turvey, J.D. Maser and J. Endicott, Multiple Recurrences of Major Depressive Disorder, *American Journal of Psychiatry* **157** (2000), 229-233.

[4] K.S. Park, W.J Yi, J.W. Seo, J.M. Choi, J.H. Han and K.H. Hong, Application of non-intrusive physiological data monitoring in normal apartment style house, *IEEE EMBS Asian-Pacific Conference on Biomedical Engineering*, (2003), 36- 37.

[5] A. Korszun, N.C. Engleberg, C.B. Brucksch, J.F. Greden and L.A. Crofford, Use of actigraphy for monitoring sleep and activity levels in patients with fibromyalgia and depression, *Journal of Psychosomatic Research*, **52** (2002), 439-443.

[6] J.F. Cohn, T. Simon, I. Matthews, Y. Yang, M. Hoai, M. Tejera, F. Zhou and F. de la Torre, Detecting depression from facial actions and vocal prosody, *Affective Computing and Intelligent Interaction and Workshops* (2009), 1-7.

[7] E. Moore, M. Clements, J. Peifer and L. Weisser, Comparing objective feature statistics of speech for classifying clinical depression, *IEMBS '04. 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (2004), 17-20.

[8] K.M. Kim S. W. Bang and S. R. Kim, Emotion Recognition System Using Short-Term Monitoring of Physiological Signals, *Medical & Biological Eng. and Computing,* **42** (2004), 419-427.

[9] M. Sung, C. Marci and A. Pentland, Objective Physiological and Behavioral Measures for Identifying and Tracking Depression State in Clinically Depressed Patients, *Technical Report Human Dynamics Group, MIT Media Laboratory* (2005).

[10] K. McBurnie, M. Kwiatkowska, L. Matthews and A. D'Angiulli, A Multi-factor Model for the Assessment of Depression Associated with Obstructive Sleep Apnea: A Fuzzy Logic Approach, *NAFIPS '07. Annual Meeting of the North American* (2007), 301-306.
[11] http://www.madep.cat/
[12] J. Martínez-Miranda and M. Lluch-Arie, Using Agents in Mental Health: A System to Support the Remote Treatment of Major Depression, *Proceedings of the 13th International Conference of the Catalan Association for Artificial Intelligence* (2010), 101-110.
[13] H. Medjahed, D. Istrate, J. Boudy and B. Dorizzi, Human activities of daily living recognition using fuzzy logic for elderly home monitoring, *FUZZ-IEEE* (2009), 2001-2006.
[14] http://muskie.usm.maine.edu/clinicalfusion/DHHS/phq9.pdf, 1999.
[15] A. Pinto-Meza, D. Suárez, X. Caseras, J.M. Haro, A. Serrano-Blanco and R. Torrubia, Evaluación telefónica de los sistemas de activación e inhibición conductual, *Revista de psiquiatría y salud mental* **2(2)**, 66-71.
[16] T. Brugha, P. Bebbington, C. Tennant and J. Hurry, The List of Threatening Experiences: a subset of 12 live event categories with considerable long-term contextual threat, *Psychological Medicine* **15** (1985), 189–194.

A First Approximation to a Cognitive Icon Query By Example Search Engine

Lledó Museros, Ismael Sanz and Zoe Falomir
Engineering and Computer Science Department
Universitat Jaume I
{museros,isanz,zfalomir}@uji.es

Abstract. The application of a qualitative shape and qualitative colour description and similarity calculus to an Image Query By Example problem is presented in this paper. Specifically, the qualitative shape and colour similarity theories are applied to the problem of matching icon images. The suitability of this approach as a foundation for a practical query by example engine for icons is shown by an experimental evaluation. The use of a more cognitive approach opens the possibility of creating systems that can be tailored to particular user requirements involving complex rules that cannot be handled by current systems.

Keywords. Icon search by example, qualitative shape similarity, qualitative colour similarity.

Introduction

For software and design professionals, finding a consistent set of icons for user interface or mapping purposes is a common and important task. The process of icon creation generally starts by a hand-drawn sketch, and includes one or more iterations in which the designer looks for source materials, or just for inspiration, in publicly available repositories in the Internet. As a result, there are many icon search engines that have been developed in order to help the designers perform this task, such as Challenger IconDB, which is a database that contains icons of various sizes and keywords attached to them, or IconLook, Icons-Search, IconSeeker, Icon Library, EasyIconFinder, Free Icons Web, Very Icon or IconsPedia which are web pages for searching and downloading icon files. All these icon search engines carry out the search by using a term or a word to describe the desired icon, instead of the actual image content of the icon as an example result. But, there are times and situations when we imagine what we desire and we are unable to express it in precise wording. For instance, a web designer may want to look for a specific type of cross symbol, but by using the term "cross" he does not get the desired result because the aesthetical aspect of the returned cross icons does not match what he has in mind. In cases such as these, a search using an example icon (or simply a scanned version of a hand-drawn sketch), as the base for the query will be very helpful and will get results closer to what the user is looking for.

Therefore, in this case, *Content-Based Image Retrieval* (CBIR) and, particularly, *Query By Example* (QBE) techniques, would be approaches better suited to retrieve useful icons. QBE is a query technique that involves providing the CBIR system with

an example image on which the search will be based. The QBE algorithms may vary depending on the application, but the result images should all share common elements with the provided example. By the nature of its task, any CBIR technology deals with two intrinsic problems: i) how to create the mathematical description of an image or *signature*; and ii) how to assess the similarity between a pair of images based on their abstracted descriptions. Sections 1 and 2 present our solution to create the icon *signatures* and section 3 presents how our solution assesses a similarity measure between a pair of icons.

There are many methods related to QBE, which usually depend on visual indexing techniques adopted from text-indexing research [1]. In Petrakis et al. [2], R-trees are used for indexing images represented as attributed relational graphs (ARGs). The retrieval of images using wavelet coefficients as image representations and R^*-trees for indexing has been studied by Natsev et al. [3]. Visual content matching using graph-based image representation and an efficient metric indexing algorithm has been proposed in Berretti et al. [4]. More details of techniques for IQBE of pictures can be found in Marsicoi et al. [5] and Del Bimbo [6]. In all these methods, the similarity between two images is made by using a signature in terms of mathematical functions capturing a certain visual property of an image, either globally for the entire image pixels or locally for a small group of pixels.

All of these approaches are optimized for dealing with photographic images; dealing with icons, however, poses some quite different problems. The biggest differences commonly associated with icons with respect to other images are their small size; the pervasive use of alphas for composition; their inherently "linear drawing" nature, with a distinctive silhouette that is emphasized at the expense or other (potentially distracting) details; and, more subtly, the fact that most icons are produced in accordance to rather strict platform- or application-specific user interface guidelines that dictate elements such as the visual characteristics of the silhouette, the preferred orientation, an uniform lighting source, and even the allowed color palette; see e.g. [7] or [8].

All of these characteristics are not specifically supported by existing QBE systems. This explains why the existing icon retrieval services completely ignore the available QBE techniques, including those that provide open source implementations that would be easy to incorporate into any system. But there is a more subtle reason: while good shape similarity retrieval, the target of most QBE systems, is a most useful primitive, it is only *a part of the actual requirements* of the user: given a set of icons with similar shapes, those that are a best fit to the concrete task of the designer (some that already include some of the features required by the target user interface guidelines, for instance) should be ranked first. However, to the best of our knowledge, no existing QBE system provides such functionality, or even attempts to address any kind of complex requirement that goes beyond pure image similarity without context.

As a consequence, in this paper we propose to deal with the problem of Icon QBE from a more cognitive point of view; this should allow capturing more complex features than those used by "pixel-oriented" systems, ultimately being powerful enough to work in cases such as the complex user requirements outlined in the previous paragraph. More importantly, a higher-level representation should be more easily *extensible* to allow quick adaptation to real-world use cases. We will address these goals by using *qualitative reasoning* techniques; we believe that by using a process closer to the one which human beings use to describe images as we can add meaning to

the qualitative concepts we use for image description and then use it for the complex reasoning required by the use case we have presented.

As Qualitative Reasoning (QR) aims to capture the fundamental aspects of a system or mechanism, while suppressing much of the detail, it is useful in situations where the information we manage is incomplete, uncertain, and even inconsistent, as in the case of shape recognition. Moreover, QR includes the study of aspects of the space, such as topological relations, which remain invariant under scale, translation and rotation transformations. Therefore, the definition and application of a theory for qualitative shape description and identification (matching) is valuable. Human beings also recognize objects qualitatively, as they can recognize a drawing of a cat just by seeing a pair of pointed ears and a furry tail. In the same way, our theory will only look for the distinguishing features of a shape in order to identify it, and it will not analyse each pixel of an image for shape identification, as is done by traditional computer vision.

Therefore, the paper presents the firsts steps of a project in which we want to develop an Icon QBE search engine by using qualitative description and matching of the shape and the colour of the icon images. Specifically we are going to apply the theory developed by Falomir et al. [9] which has been applied to the robotics field with success. By using this method the signature of an icon will not be a mathematical function but a string of symbols which describes the relevant features of the icon shape (such as its type of segments, angles, and convexity) and gives a description of the colour as a colour name. In this paper, we will demonstrate (see section 4) that the method can be used inside the general Icon QBE search engine. Moreover, by using this method we will be able to study the *quality* of an image, which can be defined at two levels: one involving concrete image parameters like size, shape, colour, etc., and the other involving high-level perception, such as those specified by user interface guidelines, which we will term as "aesthetic rules". As mentioned above, while it is relatively easy to rank images based on shape and colour, the differences may be not significant enough to use as ranking criteria. On the other hand, the use of aesthetic rules makes ranking much closer to the actual requirements of end users.

The use of a more cognitive approach for icon image description, as the one used in this paper, allows in principle the description and exploitation of aesthetic rules, in order to provide more relevant results to QBE queries and an easier adaptation to complex, real world application use cases.

The goal of this paper is to present the first step towards this vision: the implementation of the most fundamental QBE primitive, similarity retrieval, based on qualitative theories of shape and color which are used to provide high-level descriptions of icons, on which a similarity function is defined. The soundness of the approach is validated experimentally.

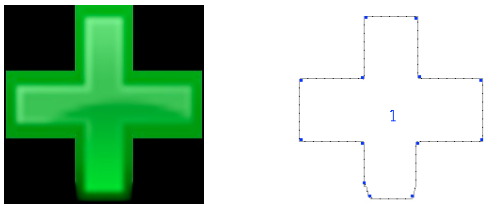
The remainder of the paper is organized as follows: Section 1 provides an overview of the qualitative theory used for shape description, followed by the theory for colour description in Section 2. In Section 3, the approach to qualitative similarity is developed. Section 4 shows experimental results, and finally Section 5 presents some conclusions and discusses the future work.

1. Overview of the Model for Qualitative Shape Description (QSD)

Our approach is based on segmenting and image and automatically extracting the boundary of any object contained within it. Then the relevant points that characterize the shape of the object (mainly vertices and points of curvature) are obtained analysing the slope defined by groups of points contained in the boundary. And, finally, each relevant point (*P*) is described by a set of features <KEC$_P$, A$_P$ or TC$_P$, L$_P$, C$_P$> defined as:

- Kind of Edges Connected (KEC) by the relevant point *P*, described as: {*line_line (l_l), line_curve (l_c), curve_line (c_l), curve_curve (c_c), curvature_point (c_p)*};
- Angle (A) at the relevant point *P*, described as: {*very_acute, acute, right, obtuse, very_obtuse*};
- Type of Curvature (TC) at the relevant point *P*, described as: {*very_acute, acute, semicircular, plane, very_plane*};
- Compared Length (L) of the two edges connected by *P*, described as: {*much_shorter (msh), half-length (hl), a_bit_shorter (absh), similar_length (sl), a_bit_longer (abl), double_length (dl), much_longer (ml)*};
- Convexity (C) at the relevant point *P*, described as: {*convex, concave*}.

Figure 1 presents the qualitative shape description of an icon composed of 13 relevant points which connect straight lines and define different angles, convexities and lengths. In the QSD the first vertex is the upper-left one.



QSD (Object) = [
1. [line-line,right,abl,convex],
2. [line-line,right,sl,convex],
3. [line-line,obtuse,sl,concave],
4. [line-line,right,sl,convex],
5. [line-line,right,sl,convex],
6. [line-line,obtuse,sl,concave],
7. [line-line,obtuse,sbl,convex],
8. [line-line,obtuse,ml,convex],
9. [line-line,very_obtuse,msh,convex],
10. [line-line,obtuse,absh,concave],
11. [line-line,right,sl,convex],
12. [line-line,right,sl,convex],
13. [line-line,obtuse,sl,concave]
].

Figure 1. QSD of an icon.

2. Overview of the Model for Qualitative Colour Description (QCD)

Our approach is based on the standard Red, Green and Blue colour channels (sRGB) of the predominant colour of the object (the mean of the sRGB colour channels of all the pixels of the image), which are translated into coordinates of Hue, Saturation and Lightness (HSL) colour space in order to give a name to the colour of the object.

From the HSL colour coordinates obtained, a reference system for qualitative colour naming is defined as $QCRS = \{UH, US, UL, QC_{LAB1..5}, QC_{INT1..5}\}$ where *UH* is the Unit of Hue; *US* is the Unit of Saturation; *UL* is the Unit of Lightness; $QC_{LAB1..5}$

refers to the colour names; and $QC_{INT1..5}$ refers to the intervals of HSL colour coordinates associated with each colour name. In our application, the QC_{LAB} and QC_{INT} are the following:

QC_{LAB1} = {black (bk), dark_ grey (dg), grey (g), light_grey (lg), white (w)}
QC_{LAB2} = {red (r), yellow (y), green (gn), turquoise (t), blue (b), purple (pu), pink (pk)}
QC_{LAB3} = {pale_ + QC_{LAB2}}
QC_{LAB4} = {light_ + QC_{LAB2}}
QC_{LAB5} = {dark_ + QC_{LAB2}}

QC_{LAB1} and QC_{LAB2} are the grey scale or the rainbow scale reference systems, respectively. The colours in the rainbow scale are considered as the strong ones, while the pale colours are given an explicit name in QC_{LAB3}. Finally, dark and light colours are given an explicit name in QC_{LAB4} and QC_{LAB5}, respectively. The intervals of HSL values which define the colour names ($QC_{INT1..5}$) have been calibrated to the images to be described, in this case the icon images. For example, the colour name given to the icon in Figure 1 is *green*.

3. Overview of the Shape and Colour Similarity Calculus

The approach to obtain dissimilarity values between qualitative parameters of shape and between qualitative colours is based on Conceptual Neighbourhood Diagrams (CNDs).

Freksa [10] determined that two qualitative terms are conceptual neighbours if "one can be directly transformed into another by continuous deformation". Therefore, acute and right angles are conceptual neighbours since an extension of the angle acute cause a direct transition to the right angle. CNDs can be described as graphs containing: (i) nodes that map to a set of individual relations defined on intervals and (ii) paths connecting pairs of adjacent nodes that represent the continuous transformations which can have weights assigned in order to establish priorities. For each of the features in our models for QSD and QCD, a CND has been defined in [9]. Then, dissimilarity matrices are constructed to map the pairs of nodes in each CND to the minimal path distance between them.

As the qualitative shape of an object is described by means of all its relevant points (RPs), in order to define a similarity measure between shapes, first a similarity between all the features that describe the relevant points has to be obtained. Hence, given two relevant points, denoted by RP_A and RP_B, belonging to the shapes of the objects A and B respectively, a similarity between them, denoted by $SimRP(RP_A, RP_B)$, is defined as:

$$SimRP(RP_A, RP_B) = 1 - \sum_{i \in \{KEC, A\vee TC, L, C\}} w_i \frac{dsShape(i)}{DsShape(i)} \quad (1)$$

where $dsShape(i)$ denotes the dissimilarity between RP_A and RP_B with respect to feature i (i ∈ KEC, A/TC, L, C}), obtained from the dissimilarity matrices in [9]. $DsShape(i)$ denotes the maximum dissimilarity in the dissimilarity matrix related to the feature i. Hence, by dividing $dsShape(i)$ and $DsShape(i)$ the proportion of dissimilarity between RP_A and RP_B related to feature i is obtained, which is between 0 and 1. The

parameter w_i is the weight assigned to feature i, and it is hold that $w_{KEC}+w_{A|TC}+w_L+w_C = 1$ and $w_i \geq 0$. The final value is substracted from 1 in order to provide a similarity between relevant points, instead of a dissimilarity.

In order to compare two shapes A and B, with n and m relevant points, respectively, the similarity between A and B ($SimQSD(A,B)$) is calculated from (1) as an arithmetic mean of the similarity value between relevant points of both shapes. If $n \geq m$, then there are some relevant points of A with no corresponding points in B. In this case, the points with no corresponding pairs are compared to the *void* relevant point and the similarity between both points is zero. The combination of comparisons of relevant points that maximizes the similarity is selected:

$$SimQSD(A,B) = \max\left(\frac{1}{n}\sum_{\substack{RP_A \in A \\ RP_B \in B}}^{m} SimRP(RP_A, RP_B)\right) \tag{2}$$

With respect to the colour similarity calculus, given two qualitative colours, denoted by QC_A and QC_B, referring to the colours of the objects A and B, a similarity between them, denoted by $SimQCD(QC_A,QC_B)$, is defined as:

$$SimQCD(QC_A, QC_B) = 1 - \frac{dsColour(QC_A, QC_B)}{MaxDsColour} \tag{3}$$

where $dsColour(QC_A,QC_B)$ denotes the dissimilarity between the colour names obtained from the dissimilarity matrices defined in [18]. *MaxDsColour* denotes the maximum dissimilarity for all colour names. Hence, by dividing $dsColour(QC_A,QC_B)$ and *MaxDsColour* the proportion of dissimilarity related to QC_A and QC_B is obtained, which is between 0 and 1. Finally, this value is substracted from 1 to provide a similarity instead of a dissimilarity. Note that $0 \leq SimQCD(QC_A, QC_B) \leq 1$ and that it is a symmetrical relation: $SimQCD(QC_A, QC_B) = SimQCD(QC_B, QC_A)$. Furthermore, this similarity is intuitive in the sense that $SimQCD(QC_A, QC_B) = 0$ means that $dsColour(QC_A, QC_B) = MaxDsColour$, that is, both colours are as different as possible (for example $SimQCD(white, black) = 0$).

Finally, a similarity measure $SimIcon(A,B)$ between two icons A and B is defined as:

$$SimIcon(A,B) = w_{QSD} * SimQSD(A,B) + w_{QCD} * SimQCD(A,B) \tag{4}$$

where the parameters w_{QSD}, and w_{QCD} are the weights assigned to the shape similarity (*SimQSD*) and the colour similarity (*SimQCD*) respectively. It must hold that $w_{QSD} + w_{QCD} = 1$. Clearly, these weights can be tuned in order to give more importance to one aspect of similarity (shape or colour) over the other.

4. Experimental Evaluation

This section presents two different experiments done with different sets of icons in order to test if the QSD, QCD and similarity calculus presented in previous sections are suitable to be used for icon query by example. The sets of icons (in .png format) chosen are named *famfamfam* [11], *nuvola* [12], and *fugue* [13].

To allow the program to segment the icon images correctly, these images have to be resized because of the small size of the icons, which causes some of their borders to be missed. The new size of the icon has to be fixed taking into account that bigger images also increases the time needed for their segmentation. Therefore all the images has been resized to 200x200, which is a size which returns good results without increasing the segmentation time unnecessarily.

The most appropriate segmentation algorithm for our problem was chosen empirically by evaluating several methods, namely Felsenwaktz and Huttenlocher's [14], Canny [15], Jseg [16], and Machine Vision Without Pixels [17]. The best performing method was Felsenwaktz and Huttenlocher's, which we used for experimentation. In most cases the segmentation parameters used are: $\sigma = 0.4$, $k = 500$ and $min = 1000$, except for more complicated images (as the ones in the second set of the second experiment) where the parameters have been $\sigma = 0.4$, $k = 750$ and $min = 2500$. These last parameters obtain better results but increase the calculus time, therefore when possible the more relaxed parameters have been used.

In order to give the same importance to the shape and to the colour features the weights w_{QSD}, and w_{QCD} of equation (4) have been established to 0.5 each. The weights of equation (1) have been established to $w_{KEC}=w_{A|TC}=w_L=w_C= 0.25$.

4.1. Icon Query By Example between icons belonging to the same set.

This section presents three experiments. In all of them the icons have been compared with icons belonging the same set as follows: i) icons in the *nuvola* set (Table 1), ii) icons in the *fugue* set (Table 2), and iii) icons in the *famfamfam* set (Table 3). The icons shown from each set have been selected in order to exemplify the method.

Tables 1 – 3 show in the first row and first column the icons being compared, and in the intersection cell they show the similarity percentage of the value obtained by applying equation (4). Note that although equation (4) returns a similarity measure between 0 and 1, the returned value has been transformed in a percentage (%) in the tables. The original size of the icons in Table 1 is 64x64, the original size of icons in Table 2 is 32x32, and the original size of icons in Table 3 is 16x16. Note that the smaller the original size of the icons is, the segmentation result is worse, but still the method obtains results good enough to distinguish between icons. For instance, in Table 1 the two red icons with a cross as symbol are more similar between them than with respect to the other icons.

Note that all the tables are symmetrical with respect to their diagonal. The tables show only 6 icon examples, but the tests have been done with all the icons of each set.

The percentages obtained show that the similarity is higher with closer colours and between figures without curves or between figures with curves.

Table 1. Qualitative Similarity Percentages between icons of the *nuvola* set.

	88	25	46	80	49	80
	25	100	51	54	40	16
	46	51	97	20	73	20
	80	54	20	90	47	66
	49	40	73	47	100	49
	80	16	20	66	49	100

Table 2. Qualitative Similarity Percentages between icons of the *fugue* set.

	96	57	76	52	25	76
	57	95	62	60	32	71
	76	62	100	58	48	66
	52	60	58	90	37	53
	25	32	48	37	90	38
	76	71	66	53	38	100

Table 3. Qualitative Similarity Percentages between icons of the *famfamfam* set.

	88	60	46	48	68	56
	60	87	30	67	48	36
	46	30	100	60	51	48
	48	67	60	86	50	44
	68	49	51	50	100	64
	56	36	48	44	64	100

4.2. Icon Query By Example between icons belonging to different sets.

This experiment shows the similarity percentage between icons belonging to different sets. Table 4 shows in the first row and column the icons being compared and in the corresponding cell the similarity percentage obtained. The icons in the row and columns number 1 and 4 are from the *famfamfam* set, rows and columns number 2 and 5 correspond to icons from the *fugue* set, and rows and columns number 3 and 6 to the *nuvola* one.

Note that the icons from the rows and columns 4,5 and 6 represent the same concept but they are from different icon sets. Their similarity is the biggest one against the rest of icons. Moreover, the similarity between the icons in row/column 4 and 6 is also bigger than with respect to the rest of icons, and as intuitive can be seen they are the one more similar in shape and colour. Therefore the method is suitable to the icon query by example.

Table 4. Qualitative Similarity Percentages between icons of three different sets.

						
	88	55	43	37	29	26
	55	90	75	36	31	31
	43	75	100	28	35	33
	37	37	28	100	70	80
	29	31	35	70	100	83
	26	31	33	80	83	100

5. Conclusions and Future Work

This paper has shown that the QSD, QCD and the similarity calculus presented in sections 2, 3, and 4 are promising approaches for calculating the similarity between icons. It serves, therefore, as proof-of-concept for the idea of using a high level qualitative representation as a basis for a practical Icon Query By Example search engine.

However, as explained in the introduction, similarity search is only a basic primitive. To be able to fully address the complex requirements that arise in real-world use cases, further research based on the foundations presented in this paper is required. Our plans for future work include:

- The extension of the theories to account for *aesthetic rules*, taking as a model common user interface guidelines. It should be possible to define different sets of rules according to application context, and to exploit them to try to achieve a higher precision in ranking that is possible using non-cognitive approaches.
- We plan to carry out cognitive tests and to use statistical inference techniques to discover common *motifs* (sequences of significant graphical elements) in representations that correspond to cognitively significant features of icons sets.

With these results, we plan to develop a learning method to automatically describe the style-related features of each icon.

- A significant strong point of traditional approaches to image QBE is the high performance achieved by the indexing possibilities given by the simple, compact representations used. The scalability of high-level, qualitative approaches such as ours depends on further research on appropriate indexing techniques.

Moreover, only the mean colour of each icon is currently considered, which does not seem to be cognitive enough; therefore, we want to improve the method by studying how to determine several predominant colours.

Finally, when the style-related features and the colour features are defined, we plan to carry out the experimental evaluation of our approach on a large database.

Acknowledgments

This work has been supported by Universitat Jaume I – Fundació Bancaixa (grants P1·1A2008-14 and P1·1B2010-4), by the Spanish National Research Program (grant TIN2008-01825/TIN). Zoe Falomir was partially supported by Pla Estrategic 2011/12 of the Department of Computer Science and Engineering of Universitat Jaume I. We are grateful to Javier Collado Ruiz for his help with the experimental part.

References

[1] Ritendra Datta, Dhiraj Joshi, Jia Li and James Z. Wang, Image Retrieval: Ideas, Influences, and Trends of the New Age, *ACM Computing Surveys, vol. **40**, no. 2, article 5 (2008)*, pp. 1-60.
[2] Petrakis, E. G. M., Faloutsos , C., and Lin, K. I., Imagemap: An image indexing method based on spatial similarity. *IEEE Trans. Knowl. Data Eng. **14**,* 5 (2002), 979–987.
[3] Natsev, A., Rastogi, R., and Shim, K.. Walrus: A similarity retrieval algorithm for image databases. *IEEE Trans. Knowl. Data Eng. **16***, 3 (2004), 301–316.
[4] Berretti, S., Bimbo, A. D., and Vicario, E. Efficient matching and indexing of graph models in content-based retrieval. *IEEE Trans. Pattern Anal. Mach. Intell. **23***, 10 (2001), 1089–1105.
[5] Marsicoi, M. D., Cinque, L., and Levialdi, S.Indexing pictorial documents by their content: A survey of current techniques. *Image Vision Comput*. ***15***, 2 (1997), 119–141.
[6] Del Bimbo, A. D. *Visual Information Retrieval*. Morgan Kaufmann, 1999.
[7] Apple User Interface Guidelines. Available at http://developer.apple.com/
[8] Windows User Experience Interaction Guidelines. Available at http://msdn.microsoft.com/
[9] Falomir Z.,Martí I.,Viana W.,Museros L.,Escrig M.T., A Pragmatic Approach for Qualitative Shape and Qualitative Colour Similarity Matching. In René Alquézar, Antonio Moreno and Josep Aguilar (Eds.), *Artificial Intelligence Research and Development, Frontiers in Artificial Intelligence and Applications*, IOS Press, Vol. 220, pp. 281-290, ISBN 978-1-60750-642-3, ISSN 0922-6389, Octubre 2010.
[10] Freksa, C., Conceptual neighbourhood and its role in temporal and spatial reasoning, In Singh, M., Trave-Massuyes, L. (eds.) *Decision Support Systems and Qualitative Reasoning*, (1991), pp. 181-187, Amsterdam.
[11] Famfamfam icon set. Available at: http://www.famfamfam.com/
[12] Nuvola icon set. Available at: http://www.icon-king.com/projects/nuvola/
[13] Fugue icon set. Available at: http://p.yusukekamiyamane.com/
[14] Felzenszwalb, P., and Huttenlocher, D. Efficient graph-based image segmentation. *International Journal of Computer Vision **59**,* 2 (2004), 167-181.
[15] Canny, J.F., A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (TPAMI), 8:679–697, 1986.
[16] Yining Deng, B. S. Manjunath, and Hyundoo Shin. Color image segmentation. In *Conference on Computer Vision and Pattern Recognition* (CVPR '99), pages 2446–2451, 1999.
[17] T. C. Folsom, *Neural Networks Modeling Cortical Cells for Machine Vision*. PhD thesis, University of Washington, 1994.

Extending the Grounded Semantics by Logic Programming Semantics

Juan Carlos NIEVES [a,1], Mauricio OSORIO [b] and Ulises CORTÉS [a]
[a] *KEMLg, Software Department, Universitat Politècnica de Catalunya, Spain*
[b] *Fundación Universidad de las Américas - Puebla, México*

Abstract. We introduce a formal argumentation method based on normal programs and rewriting systems which is able to define extensions of the grounded semantics based on specific rewriting rules which perform particular kind of reasoning as in reasoning by cases. These new argumentation semantics are intermediate argumentation semantics between the grounded and the preferred semantics.

Keywords. Argumentation Theory, Normal Logic Programs, Rewriting Systems.

Introduction

Although several approaches have been proposed for argument theory, Dung's approach presented in [7], is a unifying framework which has played an influential role on argumentation research and Artificial Intelligence (AI). Dung's approach is regarded as an abstract model where the main concern is to find the set of arguments which are considered as acceptable. The strategy for inferring the set of acceptable arguments is based on abstract argumentation semantics. The kernel of Dung's framework is supported by four abstract argumentation semantics: *stable semantics*, *preferred semantics*, *grounded semantics*, and *complete semantics*. From these semantics, the main semantics for collective acceptability are *the grounded semantics* and *the preferred semantics* [2]. The first one represents a skeptical approach, since for a given argumentation framework it always identifies a single extension, called grounded extension. The preferred semantics instead represents a credulous approach, since for a given argumentation framework it identifies a set of extensions which are called preferred extensions. It is worth mentioning that the grounded extension is included in all the preferred extensions. This property supports the fact that the grounded semantics is most adequate than the preferred semantics for developing skeptical reasoning [1]. Also we can say that the grounded semantics approach is one of the most useful argumentation approaches in real argumentation-based systems [5,9].

Since Dung's framework was introduced in [7], it was shown that this approach can be viewed as a special form of logic programming with *negation as failure*. In fact,

[1]Correspondence to: Software Department (LSI), Universitat Politècnica de Catalunya. C/Jordi Girona 1-3, E08034, Barcelona, Spain. E-mail: (J.C. Nieves: jcnieves@lsi.upc.edu), (M. Osorio: osoriomauri@googlemail.com), (U. Cortés: ia@lsi.upc.edu)

this result was used for introducing a general logic programming method for generating metainterpreters for argumentation theory.

According to Bench-Capon and Dunne [2], the principal abstract argumentation semantics introduced by Dung exhibit a variety of problems: *Emptiness*, *Non-existence* and *Multiplicity* [2]. Since the grounded semantics is sceptical semantics which always exists, the main problem of it is emptiness. The main issue of this paper is to explore some extension of the grounded semantics. In particular, we present a frame of extensions of the grounded semantics which are intermediate semantics between the grounded semantics and the preferred semantics. All these semantics have as a common point a suitable logic program and the only difference between them is the logic programming semantics which is applied to the logic program. One of the outstanding properties of these semantics is that they are polynomial time computable and they are intermediate argumentation semantics between the grounded and the preferred semantics. The logic programming semantics which are considered for extending the grounded semantics are extension of the well-founded semantics [6].

The rest of the paper is structured as follows: In §1, some concepts of argumentation theory and logic programming are presented. In §2, a transformation of an argumentation framework into a normal logic program is presented. In 3, our extensions of the grounded semantics are presented. In the last section, our conclusions are discussed.

1. Background

In this section, we first define the syntax of a valid logic program, after that we define a characterization of Well-Founded Semantics (WFS) in terms of rewriting systems and finally we present some basic concepts of argumentation theory.

1.1. Logic programs: Syntax

A signature $\mathcal{L}$ is a finite set of elements that we call atoms. A literal is an atom, a, or the negation of an atom $\neg a$. Given a set of atoms $\{a_1, \ldots, a_n\}$, we write $\neg\{a_1, \ldots, a_n\}$ to denote the set of literals $\{\neg a_1, \ldots, \neg a_n\}$. A normal clause is of the form: $a_0 \leftarrow a_1, \ldots, a_j, \neg a_{j+1}, \ldots, \neg a_n$, where a_i is an atom, $0 \leq i \leq n$. When $n = 0$ the normal clause is an abbreviation of $a_0 \leftarrow \top$, where $\top$ and$\bot$ are the ever true and ever false propositions respectively. A normal program is a finite set of normal clauses. Sometimes, we denote a clause C by $a \leftarrow \mathcal{B}^+, \neg\mathcal{B}^-$, where $\mathcal{B}^+$ contains all the positive body literals and $\mathcal{B}^-$ contains all the negative body literals. We also use $body(C)$ to denote $\mathcal{B}^+, \neg\mathcal{B}^-$. When $\mathcal{B}^- = \emptyset$, the clause C is called definite clause. A definite program is a finite set of definite clauses. We denote by $\mathcal{L}_P$ the signature of P, i.e. the set of atoms that occurs in P. Given a signature $\mathcal{L}$, we write $Prog_{\mathcal{L}}$ to denote the set of all the programs defined over $\mathcal{L}$.

A partial interpretation based on a signature $\mathcal{L}$ is a disjoint pair of sets $\langle I_1, I_2\rangle$ such that $I_1 \cup I_2 \subseteq \mathcal{L}$. A partial interpretation is total if $I_1 \cup I_2 = \mathcal{L}$. Given two interpretations $I = \langle I_1, I_2\rangle$, $J = \langle J_1, J_2\rangle$, we set $I \leq_k J$ *if, by definition,* $I_i \subseteq J_i$, $i = 1, 2$. Clearly $\leq_k$ is a partial order. When we look at interpretations as sets of literals then $\leq_k$ corresponds to $\subseteq$. A general semantics SEM is a function on $\text{Prog}_{\mathcal{L}}$ which associates with every program a partial interpretation.

Definition 1 (SEM) *For any logic program P, we define HEAD$(P) = \{a|\ a \leftarrow \mathcal{B}^+,\ \neg\mathcal{B}^- \in P\}$ — the set of all head-atoms of P. We also define SEM$(P) = \langle P^{true}, P^{false}\rangle$, where $P^{true} := \{p|\ p \leftarrow \top \in P\}$ and $P^{false} := \{p|\ p \in \mathcal{L}_P \backslash HEAD(P)\}$.*

1.2. Well-Founded Semantics

We start presenting some basic transformation rules for normal logic programs which will be considered for characterizing *WFS*.

Definition 2 (Basic Transformation Rules) *[6] A transformation rule is a binary relation on Prog$_{\mathcal{L}}$. The following transformation rules are called* basic*. Let a program $P \in$ Prog$_{\mathcal{L}}$ be given.*

RED$^+$: *This transformation can be applied to P, if there is an atom a which does not occur in HEAD(P). **RED**$^+$ transforms P to the program where all occurrences of $\neg a$ are removed.*

RED$^-$: *This transformation can be applied to P, if there is a rule $a \leftarrow \top \in P$. **RED**$^-$ transforms P to the program where all clauses that contain $\neg a$ in their bodies are deleted.*

Success: *Suppose that P includes a fact $a \leftarrow \top$ and a clause $q \leftarrow body$ such that $a \in body$. Then we replace the clause $q \leftarrow body$ by $q \leftarrow body \setminus \{a\}$.*

Failure: *Suppose that P contains a clause $q \leftarrow body$ such that $a \in body$ and $a \notin HEAD(P)$. Then we erase the given clause.*

Loop: *We say that P_2 results from P_1 by* Loop$_A$ *if, by definition, there is a set A of atoms such that 1. for each rule $a \leftarrow body \in P_1$, if $a \in A$, then $body \cap A \neq \emptyset$, 2. $P_2 := \{a \leftarrow body \in P_1 | body \cap A = \emptyset\}$, 3. $P_1 \neq P_2$.*

Let CS_0 be the rewriting system such that contains the transformation rules: RED^+, RED^-, $Success$, $Failure$, and $Loop$. We denote the uniquely determined normal form of a program P with respect to the system $\mathcal{CS}$ by $norm_{\mathcal{CS}}(P)$. Every system $\mathcal{CS}$ induces a semantics SEM$_{\mathcal{CS}}$ as follows: SEM$_{\mathcal{CS}}(P) :=$ SEM$(norm_{\mathcal{CS}}(P))$.

WFS is one of the most acceptable semantics in logic programming. It was introduced in [10] and was characterized in terms of rewriting systems in [3]. This characterization is defined as follows:

Lemma 1 *[3] CS_0 is a confluent rewriting system. It induces a 3-valued semantics that it is the Well-founded Semantics.*

1.3. Argumentation theory

Now, we define some basic concepts of Dung's argumentation approach. The first one is an argumentation framework. An argumentation framework captures the relationships between the arguments (All the definitions of this subsection were taken from the seminal paper [7]).

Definition 3 *An argumentation framework is a pair $AF := \langle AR, attacks\rangle$, where* AR *is a finite set of arguments, and* attacks *is a binary relation on* AR, i.e. attacks $\subseteq AR \times AR$. *We write $\mathcal{AF}_{AR}$ to denote the set of all the argumentation frameworks defined over AR.*

We say that *a attacks b* (or b is attacked by a) if $attacks(a, b)$ holds. Similarly, we say that a set S of arguments attacks b (or b is attacked by S) if b is attacked by an argument in S.

Definition 4
- *A set S of arguments is said to be conflict-free if there are no arguments A, B in S such that A attacks B.*
- *An argument $A \in AR$ is acceptable with respect to a set S of arguments if and only if for each argument $B \in AR$: If B attacks A then B is attacked by S*
- *A conflict-free set of arguments S is admissible if and only if each argument in S is acceptable w.r.t. S.*

The (credulous) semantics of an argumentation framework is defined by the notion of preferred extensions.

Definition 5 *A preferred extension of an argumentation framework AF is a maximal (w.r.t. inclusion) admissible set of AF.*

The grounded semantics is defined in terms of a *characteristic function*.

Definition 6 *The characteristic function, denoted by F_{AF}, of an argumentation framework $AF = \langle AR, attacks \rangle$ is defined as follows:*
$F_{AF} : 2^{AR} \rightarrow 2^{AR}$
$F_{AF}(S) = \{A |$ *A is acceptable w.r.t. S* $\}$

Definition 7 *The grounded extension of an argumentation framework AF, denoted by GE_{AF}, is the least fixed point of F_{AF}*

2. Mapping from argumentation frameworks to normal programs

In order to see an argumentation framework as a normal program, we introduce a mapping from an argumentation framework to a normal logic program. This mapping was introduced in [11].

In our mapping, we use the predicate $d(X)$, where the intended meaning of $d(X)$ is "X is a defeated argument". Also we will denote by $D(A)$ the set of arguments that directly attack the argument A[2]. We define a transformation function *w.r.t.* an argument as follows.

Definition 8 *Let $AF := \langle AR, Attacks \rangle$ be an argumentation framework and $A \in AR$. We define the transformation function $\Psi(A)$ as follows:*

$$\Psi(A) := (\bigcup_{B \in D(A)} d(A) \leftarrow \neg d(B)) \cup (\bigcup_{B \in D(A)} d(A) \leftarrow \bigwedge_{C \in D(B)} d(C))$$

The direct generalization of the transformation function Ψ to an argumentation framework is defined as follows:

[2]Given AF = $\langle AR, Attacks \rangle$ and $A \in AR$. $D(A) := \{B | (B, A) \in Attacks\}$.

Definition 9 *Let $AF := \langle AR, Attacks\rangle$ be an argumentation framework. We define its associated normal program as follows: $\Psi_{AF} := \bigcup_{A\in AR} \Psi(A)$.*

3. WFS' extensions and the grounded semantics

In this section, we present the main results of our paper. In particular, we introduce a set of extensions of the grounded semantics by using a set of extensions of the well-founded semantics.

We start by presenting some basic terms. Given an argumentation framework $AF := \langle AR, Attacks\rangle$, we understand $f(E) := \{d(a)|a \in E\}$, where $E \subseteq AR$.

In [4], it was showed that Ψ_{AF} and *WFS* characterize the grounded semantics as follows:

Lemma 2 *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework and $S \subseteq AR$. S is the grounded extension of AF if and only if $\exists\ D \subseteq AR$ such that $\langle f(D), f(S)\rangle$ is the well-founded model of Ψ_{AF}.*

Since *WFS* is a 3-valued logic semantics, where any atom could be *true*, *false*, and *undefined*, we will define the concept of a 3-valued extension, where any argument could be *accepted*, *defeated*, and *undecided*.

Definition 10 (3-valued extension) *Given an argumentation framework $AF := \langle AR, attacks\rangle$, and $S, D \subseteq AR$. A 3-valued extension is a tuple $\langle S, D\rangle$, where $S \cap D = \emptyset$ and S is a conflict-free set. We call an argument a acceptable if $a \in S$, an argument b defeated if $b \in D$, and an argument c undecided if $c \in AR \setminus \{S \cup D\}$.*

3.1. $WFS^{LLC'}$ semantics

The first WFS' extension that we will consider is called $WFS^{LLC'}$ and is based on the transformation rule LLC' (Local Logic Consequence).

Definition 11 (LLC') *[6] Let a be an atom that occurs negatively in a program P and also appears in the head of some rule. Let P_1 be the program that results from P by removing $\neg a$ from every clause of P. Let **Success*** denote the reflexive and transitive closure of the relation **Success**. Suppose that P_1 relates to P_2 by **Success*** and $a \in P_2$. In this case, we add $a \leftarrow \top$ to P.*

By considering the transformation rule LLC', it is defined the rewriting system CS_1 as follows: $CS_1 := CS_0 \cup \{LLC'\}$. $WFS^{LLC'}$ is defined as follows:

Lemma 3 *[6] CS_1 is a confluent rewriting system. It induces a 3-valued semantics that we call $WFS^{LLC'}$.*

Now by considering $WFS^{LLC'}$, it is introduced an extension of the grounded semantics.

Definition 12 *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework and $S, D \subseteq AR$. $\langle S, D\rangle$ is the $WFS^{LLC'}$-extension of AF if and only if $\langle f(D), f(S)\rangle$ is a $WFS^{LLC'}$- model of Ψ_{AF}.*

The main difference between the grounded extension and the $WFS^{LLC'}$-extension is done by the transformation rule **LLC'**. Based on Ψ_{AF}, we can say that LLC' first removes all the attacks of an argument a from AF; therefore it is reviewed by *Success* whether the argument a is defeated. In case that a appears defeated, it will be assumed that the argument a is defeated. Notice that the only case that a could be defeated after removed its attacks is that a belongs to a cycle of attacks. Let us consider the following example.

Example 1 *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework, where AR :=* $\{a, b, c\}$ *and attacks :=* $\{(a, a), (a, b), (b, c), (c, b)\}$. *Hence, Ψ_{AF} is:*

$$\begin{array}{lll} d(a) \leftarrow \neg d(a). & d(a) \leftarrow d(a). & d(c) \leftarrow \neg d(b). \\ d(b) \leftarrow \neg d(a). & d(b) \leftarrow \neg d(c). & d(c) \leftarrow d(c), d(a). \\ d(b) \leftarrow d(a). & d(b) \leftarrow d(c). & \end{array}$$

To infer the AF's $WFS^{LLC'}$-extension, we need to get the Ψ_{AF}'s $WFS^{LLC'}$ model. Then, we apply CS_1 to Ψ_{AF}. We can see that the argument a *is a* controversial argument *since it is attacked by itself. Then the transformation rule LLC', first it will remove all the atoms of the form $\neg d(a)$. This means that, it will remove all the* a*'s attacks of AF. After that, it will view if* a *is defeated. Since* a *appears defeated, it is assumed that the argument* a *is a defeated argument and it is added this assumption $(d(a) \leftarrow \top)$ to the program Ψ_{AF}*

$$\begin{array}{lll} d(a) \leftarrow \neg d(a). & d(a) \leftarrow d(a). & d(c) \leftarrow \neg d(b). \\ d(b) \leftarrow \neg d(a). & d(b) \leftarrow \neg d(c). & d(c) \leftarrow d(c), d(a). \\ d(b) \leftarrow d(a). & d(b) \leftarrow d(c). & d(a) \leftarrow \top. \end{array}$$

If we assume that a *is a defeated argument, then RED^- will remove all its attacks (the clauses which will be removed are: $d(a) \leftarrow \neg d(a)$ and $d(b) \leftarrow \neg d(a)$) and $Success$ will remove all its supports to other arguments (the clause $d(c) \leftarrow d(c), d(a)$ is reduced to $d(c) \leftarrow d(c)$).*

$$\begin{array}{lll} d(a) \leftarrow d(a). & d(b) \leftarrow \neg d(c). & d(c) \leftarrow \neg d(b). \\ d(b) \leftarrow d(a). & d(b) \leftarrow d(c). & d(c) \leftarrow d(c). \\ d(a) \leftarrow \top. & & \end{array}$$

Then applying $Success$, it is found that the argument b *is a defeated argument:*

$$\begin{array}{lll} d(a) \leftarrow \top. & d(b) \leftarrow \neg d(c). & d(c) \leftarrow \neg d(b). \\ d(b) \leftarrow \top. & d(b) \leftarrow d(c). & d(c) \leftarrow d(c). \end{array}$$

Therefore applying RED^-, it removes all the attacks of the argument b*:*

$$\begin{array}{lll} d(a) \leftarrow \top. & d(b) \leftarrow \neg d(c). & \\ d(b) \leftarrow \top. & d(b) \leftarrow d(c). & d(c) \leftarrow d(c). \end{array}$$

Since the attack of the argument b *to* c *is removed, Loop will remove the clause $d(c) \leftarrow d(c)$. Then we get:*

$$d(b) \leftarrow \top. \qquad d(a) \leftarrow \top. \qquad d(b) \leftarrow \neg d(c).$$

Finally, since the argument b *was already fixed as a defeated argument,* RED^+ *will remove the attack of the argument* c *to* b *which is represented by the clause:* $d(b) \leftarrow \neg d(c)$. *Then, the* normal form *of* Ψ_{AF} *is:*

$$d(b) \leftarrow \top. \qquad\qquad d(a) \leftarrow \top.$$

Therefore, $WFS^{LLC'}(\Psi_{AF}) := \langle \{d(a), d(b)\}, \{d(c)\} \rangle$, *this means that* $\langle \{c\}, \{a, b\} \rangle$ *is the* AF*'s* $WFS^{LLC'}$ *- extension. We can conclude that the argument* c *is an acceptable argument and* a, b *are defeated arguments. Notice that* AF *has an empty grounded extension,* AF *has no stable extensions and* AF *has only one preferred extension which is* $\{c\}$. *In fact, the set of acceptable arguments of the* $WFS^{LLC'}$ *- extension corresponds to the only preferred extension of* AF.

3.2. WFS^{WK} *semantics*

Now, let us consider another extension of WFS which is called WFS^{WK}. This semantics is based on the transformation rule *Weak-Cases* which is defined as follows:

Definition 13 (Weak-Cases) *Let* P *be a program and suppose the following condition holds:* $C_1 \in P$, $C_2 \in P$, C_1 *is of the form* $a \leftarrow l$ *and* C_2 *is of the form* $a \leftarrow \neg l$. *Then the Weak-Cases transformation replaces the clauses* C_1 *and* C_2 *in* P *by the single clause* $a \leftarrow \top$.

Let CS_2 be a rewriting system which contains the transformation rules $CS_0 \cup \{$*Weak-Cases* $\}$. Then, WFS^{WK} is defined as follows:

Lemma 4 CS_2 *is a confluent rewriting system. It induces a 3-valued semantics that we call* WFS^{WK}.

Since *Weak-Cases* is an instance of the transformation rule *T-Weak-Cases*, which is defined in [6], this lemma is straightforward from Theorem 7.11 of [6].

Now, by considering WFS^{WK} semantics, it is defined another extension of the grounded semantics.

Definition 14 *Let* $AF := \langle AR, attacks \rangle$ *be an argumentation framework and* $S, D \subseteq AR$. $\langle S, D \rangle$ *is the* WFS^{WK}*-extension of AF if and only if* $\langle f(D), f(S) \rangle$ *is a* WFS^{WK}*-model of* Ψ_{AF}.

The main difference between the characterizations of the grounded semantics and the WFS^{WK}-extension is made by the transformation rule *Weak-Cases*. It is worth mentioning that essentially the transformation rule *Weak-Cases* deploys a reasoning by cases. In order to illustrate the WFS^{WK}-extension, let us consider the following example.

Example 2 *Let* $AF := \langle AR, attacks \rangle$ *be an argumentation framework, where* $AR := \{a, b, c, d\}$ *and* $attacks := \{(a, b), (b, a), (a, c), (b, c), (c, d)\}$. *Then,* Ψ_{AF} *is:*

$$
\begin{array}{lll}
d(a) \leftarrow \neg d(b). & d(a) \leftarrow d(a). & d(d) \leftarrow \neg d(c). \\
d(b) \leftarrow \neg d(a). & d(b) \leftarrow d(b). & d(d) \leftarrow d(b), d(a). \\
d(c) \leftarrow \neg d(b). & d(c) \leftarrow d(b). & \\
d(c) \leftarrow \neg d(a). & d(c) \leftarrow d(a). &
\end{array}
$$

In order to infer the WFS^{WK}-extension of AF, it is applied CS_2 to Ψ_{AF}. First of all, we can see that the argument a *is controversial w.r.t. the argument* c *because* a *is attacking to* c $(d(c) \leftarrow \neg d(a))$ *and also* a *is defending to* c $(d(c) \leftarrow d(a))$. *Therefore, if* a *is fixed as an acceptable argument, then* c *will be a defeated argument. Moreover, if* a *is fixed as a defeated argument, then* c *also will be a defeated argument. Under this situation, the transformation rule* Weak-Cases *will assume that the argument* c *is defeated, then it will remove the clauses $d(c) \leftarrow \neg d(a)$ and $d(c) \leftarrow d(a)$ from Ψ_{AF} and the clause $d(c) \leftarrow \top$ is added to Ψ_{AF}. Notice that the argument* b *is also controversial w.r.t.* c. *Then the clauses $d(c) \leftarrow \neg d(b)$ and $d(c) \leftarrow d(b)$ are removed from Ψ_{AF}.*

$$
\begin{array}{lll}
d(a) \leftarrow \neg d(b). & d(a) \leftarrow d(a). & d(d) \leftarrow \neg d(c). \\
d(b) \leftarrow \neg d(a). & d(b) \leftarrow d(b). & d(d) \leftarrow d(b), d(a). \\
d(c) \leftarrow \top. & &
\end{array}
$$

Since the argument c *was assumed as to be a defeated argument, the RED^{-} will remove* c*'s attacks. Hence, we get:*

$$
\begin{array}{lll}
d(a) \leftarrow \neg d(b). & d(a) \leftarrow d(a). & d(d) \leftarrow d(b), d(a). \\
d(b) \leftarrow \neg d(a). & d(b) \leftarrow d(b). & d(c) \leftarrow \top.
\end{array}
$$

Since this program is the formal form of Ψ_{AF}, $WFS^{WK}(\Psi_{AF}) := \langle \{d(c)\}, \{\}\rangle$. Hence $\langle \{\}, \{c\}\rangle$ is the WFS^{WK}-extension of AF. This means that the argument c is defeated.

Notice that the grounded extension of AF is the empty set, there are two stable extensions which are $\{a, d\}$ and $\{b, d\}$, and there are two preferred extensions which coincide with the stable extensions: $\{a, d\}$ and $\{b, d\}$. It is worth mentioning that usually any argument which does not belong to a preferred/stable extension is considered defeated. Then we can see that both preferred/stable extensions of AF coincide that the argument c is a defeated argument. Therefore we can appreciate that the WFS^{WK}-extension coincides with the preferred/stable extensions that the argument c is defeated.

3.3. $WFS^{WK+LCC'}$ semantics

We have defined two extensions of the grounded semantics based on two extensions of WFS, where the main support of these extensions is the use of the transformation rules: $Weak - Cases$ and LLC'. Now the combination of these transformation rules also suggests another extension of the grounded semantics.

Let $CS_3 := CS_0 \cup \{LLC', \textit{Weak-Cases}\}$. Obviously, CS_3 also defines an extension of WFS which is defined as follows:

Lemma 5 *CS_3 is a confluent rewriting system. It induces a 3-valued semantics that we call $WFS^{WK+LLC'}$.*

Then by considering $WFS^{WK+LLC'}$, we define an extension of the grounded semantics.

Definition 15 *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework and $S, D \subseteq AR$. $\langle S, D\rangle$ is the $WFS^{WK+LLC'}$- extension of AF if and only if $\langle f(D), f(S)\rangle$ is a $WFS^{WK+LLC'}$- model of Ψ_{AF}.*

None of both $WFS^{LLC'}$ and WFS^{WK} extensions is the same to $WFS^{WK+LLC'}$-extension. In order to illustrate this difference let us consider the following example.

Example 3 *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework, where $AR := \{a, b, c, d, e,\ f, m, n, p\}$ and $attacks := \{(a, b),\ (b, c),\ (c, a),\ (a, d),\ (d, e),\ (e, f),\ (m, e),\ (n, m),\ (n, p), (p, m),\ (p, n)\}$. It is not difficult to see that $WFS^{LLC'}$-extension $:= \langle\{\}, \{a, b, c, d, e\}\rangle$, WFS^{WK}-extension $:= \langle\{\}, \{m\}\rangle$, $WFS^{WK+LLC'}$-extension $:= \langle\{\}, \{a, b, c, d, e, m\}\rangle$, and the grounded extension is empty.*

This argumentation framework has no stable extensions and has two preferred extensions: $\{n\}$ and $\{p\}$.

3.4. Formalizing the extensions of the grounded semantics

Once we have defined a direct relationship between abstract argumentation semantics and logic programming semantics, it is possible to understand the behavior of some abstract argumentation semantics based on the properties of the logic programming semantics. For instance, since the grounded semantics is characterized by Ψ_{AF} and *WFS*, we can infer that the $WFS^{LLC'}$-extension, the WFS^{WK}-extension and the $WFS^{WK+LLC'}$-extension are extensions of the grounded semantics and are polynomial time computable. This is essentially because the semantics $WFS^{LLC'}$, WFS^{WK} and $WFS^{WK+LLC'}$ are extensions of *WFS* and are polynomial time computable. This result is formalized with the following theorem:

Theorem 1 *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework and E be the grounded extension of AF. Then*

a) 1. *If $\langle S, D\rangle$ is the $WFS^{LLC'}$-extension of AF then $E \subseteq S$.*
2. *If $\langle S, D\rangle$ is the WFS^{WK}-extension of AF then $E \subseteq S$.*
3. *If $\langle S, D\rangle$ is the $WFS^{WK+LLC'}$-extension of AF then $E \subseteq S$.*

b) 1. *The $WFS^{LLC'}$-extension of AF is polynomial time computable.*
2. *The WFS^{WK}-extension of AF is polynomial time computable.*
3. *The $WFS^{WK+LLC'}$-extension of AF is polynomial time computable.*

Another property that can be formalized *w.r.t.* the new argumentation semantics is that they are intermediate logic between the grounded semantics and the preferred semantics. This is essentially because the semantics $WFS^{LLC'}$, WFS^{WK} and $WFS^{WK+LLC'}$ are stronger than *WFS* and weaker than the pstable semantics (the formal definition of pstable semantics is presented in [12]). Remember that the pstable models of Ψ_{AF} correspond to the preferred extensions of AF [4].

In order to show that our new argumentation semantics are intermediate semantics between the grounded semantics and the preferred semantics, we will show that they are contained in the preferred semantics.

Theorem 2 *Let $AF := \langle AR, attacks \rangle$ be an argumentation framework, E be a preferred extension of AF, and $E' := AR \setminus E$. Then,*

1. *If $\langle S, D \rangle$ is the $WFS^{LLC'}$-extension of AF then $S \subseteq E$ and $D \subseteq E'$.*
2. *If $\langle S, D \rangle$ is the WFS^{WK}-extension of AF then $S \subseteq E$ and $D \subseteq E'$.*
3. *If $\langle S, D \rangle$ is the $WFS^{WK+LLC'}$-extension of AF then $S \subseteq E$ and $D \subseteq E'$.*

4. Conclusions and Future Work

Our experience in the interaction between argumentation semantics and logic programming semantics suggests that the correct understanding of the behavior of one side helps to understand the behavior of the other side. For instance, thanks to the deep study that there is on the well-founded semantics is easy to understand the behavior of any extension of the grounded semantics which is based on an extension of the well-founded semantics.

In particular in this paper, we showed that by using extension of the well-founded semantics, it is possible to define an intermediate reasoning between the grounded semantics and the preferred semantics.

A fundamental step in our future research is to explore the relation between the extensions of the grounded semantics introduced in this paper and the ideal semantics [8]. In fact, we can see that at least with the examples of this paper and the new extensions of the grounded semantics coincide with the ideal semantics.

References

[1] P. Baroni and M. Giacomin. Evaluating argumentation semantics with respect to skepticism adequacy. In *ECSQARU 2005, Barcelona, Spain, July 6-8, 2005, Proceedings*, volume 3571 of *LNCS*, pages 329–340. Springer, 2005.

[2] T. J. M. Bench-Capon and P. E. Dunne. Argumentation in artificial intelligence. *Artificial Intelligence*, 171(10-15):619–641, 2007.

[3] S. Brass, U. Zukowski, and B. Freitag. Transformation-based bottom-up computation of the well-founded model. In *NMELP*, pages 171–201, 1996.

[4] J. L. Carballido, J. C. Nieves, and M. Osorio. Inferring Preferred Extensions by Pstable Semantics. *Iberoamerican Journal of Artificial Intelligence (Inteligencia Artificial) ISSN: 1137-3601, (doi: 10.4114/ia.v13i41.1029)*, 13(41):38–53, 2009.

[5] C. I. Chesñevar, A. G. Maguitman, and R. P. Loui. Logical models of argument. *ACM Comput. Surv.*, 32(4):337–383, 2000.

[6] J. Dix, M. Osorio, and C. Zepeda. A general theory of confluent rewriting systems for logic programming and its applications. *Ann. Pure Appl. Logic*, 108(1-3):153–188, 2001.

[7] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.

[8] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(issues 10-15):642–674, 2007.

[9] A. J. García and G. R. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1-2):95–138, 2004.

[10] A. V. Gelder, K. A. Ross, and J. S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–650, 1991.

[11] J. C. Nieves, M. Osorio, and U. Cortés. Preferred Extensions as Stable Models. *Theory and Practice of Logic Programming*, 8(4):527–543, July 2008.

[12] M. Osorio, J. A. Navarro, J. R. Arrazola, and V. Borja. Logics with Common Weak Completions. *Journal of Logic and Computation*, 16(6):867–890, 2006.

Argumentation-based Negotiation in t-DeLP-POP[1]

Pere PARDO [a,2], Pilar DELLUNDE [a,b] and Lluís GODO [a]
[a] *IIIA-CSIC, Institut d'Investigació en Intel·ligència Artificial*
[b] *Universitat Autònoma de Barcelona*

Abstract. In this contribution, we propose to model argumentation-based negotiation in terms of t-DeLP-POP, a partial order planning system that incorporates temporal defeasible logic. This logic combines temporal facts and durative rules into temporal arguments. We propose a dialogue protocol for the negotiation of plans in this planning system that models a variety of scenarios for argumentative negotiation of complex services. Then we consider case studies in the literature can be naturally modeled by dialogues in this logic-based planning framework.

Keywords. Negotiation, Argumentation, Planning, Temporal Defeasible Logic.

1. Introduction

Negotiation skills are an important ability for autonomous agents in decentralized multi-agent systems. Agents pursuing their own goals are inter-dependent, specially if abilities, rights or resources are unequally distributed among them.

Traditionally, purely quantitative approaches to negotiation are somewhat limited in expressivity, since agents merely exchange offers and accept/reject messages, instead of communicating what is good or bad with an offer, and why. Argumentation-based negotiation (ABN) is a recent area of research that tries to overcome these limitations along this line and speed up the process of reaching agreements.

In this contribution we study a logic-based planning framework as a foundation for ABN, with a focus, as in [14], on the agent architecture. The present framework aims at a two-fold integration of argumentation within the mental model of the negotiating agent: descriptive argumentation is based on the *logical program* of an agent (i.e. on her beliefs: facts and rules; while practical argumentation builds upon the *planning domain* of some agent (beliefs, actions and goals). An agent's plan search also involves taking part in several concurrent, pairwise ABN dialogues with other agents. Communicated content, though, may propagate to dialogues involving other agents. In any case, communications are determined by the agents' beliefs and concession functions.

[1]The authors acknowledge partial support of the Spanish MICINN project CONSOLIDER-INGENIO 2010 Agreement Technologies CSD2007-00022; LoMoReVI FFI2008-03126-E/FILO (FP006); ARINF TIN2009-14704-C03-03 and the Generalitat de Catalunya grant 2009-SGR-1434.

[2]Corresponding Author: P. Pardo, IIIA-CSIC, Campus UAB s/n, Bellaterra, Catalonia (Spain); E-mail: pardo@iiia.csic.es

A negotiation dialogue starts once a customer communicates some of her goals (say, *be at Oslo at* $t + 5$). The t-DeLP-POP planning system models *negotiable services* as actions that can be used to solve these goals by refining the customer's plans with them (e.g. *buy train tickets to Oslo at* t). Thus, planning actions are the atomic objects of negotiation and offers consist of a pair of such actions to be exchanged. Arguments are also ways to enforce an open goal (i.e. as the argument's conclusion), just like actions (as one of its effects), and as such, they can form as an offer t-DeLP arguments represent causal or temporal processes, with premises and conclusion being, resp., causes and effect. These processes may become triggered -intentionally or not- by the execution of planned actions (plus initial facts). Thus, within a negotiation it can be argued whether the plan-triggered processes will actually succeed w.r.t. the goals (e.g. the expected process from *being at the train to Oslo at* $t + 1$ to *be at Oslo at* $t + 5$ can be threatened by a *snow storm occurred at* t).

Speech acts of ABN dialogues between negotiating agents consist of:

- proposals (demands; offers or acceptance), encoded as goals, resp., as plan steps,
- theoretical arguments (plan threats, agent threats), encoded as argument threats,
- practical arguments (persuasion; challenge), as actions; resp., planning domains.

The present t-DeLP-POP-based model of ABN makes use of the idealized assumption: agents are totally honest and trust each other.[3]

The paper is structured as follows: we briefly introduce t-DeLP-POP in the Preliminaries, and discuss basic issues in its multi-agent extension. Then we present the negotiation framework and the protocol for ABN dialogues. Finally, we model and discuss in our framework some examples from the ABN literature.

Related Work. Our approach is a temporal extension of defeasible logic DeLP [6], and its combination with partial order planning (POP) in [7]. A multi-agent extension of the latter was studied in [10] for the cooperative case. Here we explore a more general case: including non-cooperative scenarios too, and based on the temporal defeasible logic t-DeLP [11], and its combination with POP [9].

In the literature, some proposals for ABN exist based on general argumentation frameworks, of argumentation for ABN [2,3,5], (that also include agent threats). For ABN with discussion of goals and services, proposals based on some modal (epistemic, dynamic) logic exist, like [16], [8] or the (multi-context) BDI logic [14]; see also [15], [4] for negotiation protocols). While the modal logics can express nesting of mental attitudes (beliefs, etc.,) they also present limitations due to monotonicity (a less convenient representation of actions).

2. Preliminaries: t-DeLP-POP

t-DeLP is a temporal extension of the DeLP argumentation framework proposed in [11]. We take $\mathbb{N}$ as our working set of discrete time-points. The language consists of temporal literals and rules. Temporal literals are of the form $\ell = \langle p, t\rangle$ or $\ell = \langle \sim p, t\rangle$ and express "p (resp. not-p) holds at time t", from a given set of variables $p \in \mathsf{Var}$. Strong negation,

[3] Thus, relevant information, i.e. arguments, are freely given regardless of the consequences w.r.t. one's interests. The necessary additions or modifications of the present approach lie out of the scope of this paper.

denoted $\sim p$, extends to literals: $\sim\ell = p$ if $\ell = \sim p$ and $\sim\ell = \sim p$ if $\ell = p$; and also to sets: $\overline{X} = \{\sim\ell \mid \ell \in X\}$. Defeasible rules δ are of the form $\langle p, t\rangle \multimap \langle p_0, t_0\rangle, \ldots, \langle p_n, t_n\rangle$, satisfying $t \geq \max\{t_0, \ldots, t_n\}$. Such rules read: premises $\langle p_0, t_0\rangle, \ldots$ constitute in principle a cause for (or a reason for) $\langle p, t\rangle$; thus, $\mathsf{body}(\delta) := \{\langle p_0, t_0\rangle, \ldots\}$ precede or occur no later than its conclusion $\mathsf{head}(\delta) := \langle p, t\rangle$. A special type of rules are *persistence* rules δ_p of the form $\langle p, t+1\rangle \multimap \langle p, t\rangle$, stating p in principle will be preserved from t to $t+1$. Knowledge of p *holds at* t is represented as a strict rule with empty body $\langle p, t\rangle \leftarrow$ in the set of initially known facts Ψ, but will be denoted simply as $\langle p, t\rangle$.

A temporal defeasible logic program (t-de.l.p.) is a pair (Ψ, Δ) where Ψ is a consistent set of basic facts (i.e. no pair $\langle p, t\rangle$, $\langle \sim p, t\rangle$ is in Ψ) and Δ is a set of temporal defeasible rules. The notion of *derivability of literals* in a t-de.l.p. (Ψ, Δ) is defined by closure of Ψ under *modus ponens* with Δ rules.

Derivability is monotonic and, typically, the literals derivable in a program will form form an inconsistent set, i.e. with some p and $\sim p$ being derivable.[4] The refined, consistent notion of (non-monotonic) logical consequence (called *warrant*) is defined by means of an argumentative process.

An *argument* for a temporal literal $\langle p, t\rangle$ is a $\subseteq$-minimal set of rules $\mathcal{A} \subseteq \Delta$ such that $\mathcal{A} \cup \Psi$ is consistent and that $\langle p, t\rangle$ is derivable from $\mathcal{A} \cup \Psi$ -the latter also denoted $\mathsf{concl}(\mathcal{A}) = \langle p, t\rangle$; while its set of premises is defined by $\mathsf{base}(\mathcal{A}) := \mathsf{body}[\mathcal{A}] \setminus \mathsf{head}[\mathcal{A}]$ and the argument's duration $\|\mathcal{A}\|$ is the difference $t - t'$, where t' is given by the earliest of $\mathcal{A}$'s premises $\langle p', t'\rangle \in \mathsf{base}(\mathcal{A})$. An argument $\mathcal{B}$ for another literal $\langle q, t'\rangle$ is a subargument of $\mathcal{A}$ whenever $\mathcal{B} \subseteq \mathcal{A}$ and $\mathsf{base}(\mathcal{B}) \subseteq \mathsf{base}(\mathcal{A})$; this $\mathcal{B}$ is denoted $\mathcal{A}(\langle q, t'\rangle)$.

An argument $\mathcal{A}$ for $\langle p, t\rangle$ *attacks* another argument $\mathcal{A}'$ when $\mathcal{A}'$ has a subargument $\mathcal{B}$ for $\langle \sim p, t\rangle$. Attacks merely point out the existence of a logical conflict. A preference relation between arguments is therefore needed to decide which arguments prevail. In this temporal framework, we opt for a purely formal criterion of preference for *better information*, based on more information in the premises, or based on information which is more recent (i.e. temporally closer to the time t of the attacked $\mathsf{concl}(\mathcal{B}) = \langle \sim p, t\rangle$).

Finally, a literal $\langle \ell, t\rangle$ is *warranted* in (Ψ, Δ), denoted $\ell \in \mathsf{warr}(\Psi, \Delta)$, if an argument $\mathcal{A}$ for $\langle \ell, t\rangle$ exists and is undefeated by the other arguments. To see whether this is the case, first note defeaters can have defeaters (see Figure 1 (Top Left)); the relation *is defeated by* determines a tree with root $\mathcal{A}$. A marking procedure from leaf nodes (unattacked arguments are undefeated) up to $\mathcal{A}$ is defined by the condition: $\mathcal{B}$ is defeated iff it is defeated by an undefeated argument $\mathcal{C}$ in the tree (see Figure 1 (Bottom Left)).

A basic property of t-DeLP is that it prunes inconsistencies out of any t-de.l.p.:

Theorem 1 *[11] For any t-de.l.p.* (Ψ, Δ), $\mathsf{warr}(\Psi, \Delta)$ *is consistent.*

t-DeLP-POP [9] is a planning system that combines backward search in partial order planning (POP) with forward reasoning by means of t-DeLP to compute plan progression. A feature of this system is that the representation of a an action splits into a deterministic action and an associated "non-monotonic theory", encoded within Δ, capturing effects that are context-dependent[5].

[4] Following the example in section 1, the basic facts (*be at the train to Oslo at* $t+1$ and *a snow storm occurred at* t) suffice to trigger a rule concluding that *[I will] be at Oslo at* $t+5$ and a rule for its negation.

[5] Thus we split a classical planning action *go by train to Oslo* into: (1) an action $\mathsf{get.on.train.Oslo(t)}$ with effect $\langle \mathsf{got.on.train.to.Oslo, t}\rangle$; (2) a rule from this effect to $\langle \mathsf{be.at.Oslo, t+4}\rangle$, and an exception rule for its negation further based on the the fact $\langle \mathsf{snow.storm.at.Oslo, t}\rangle$.

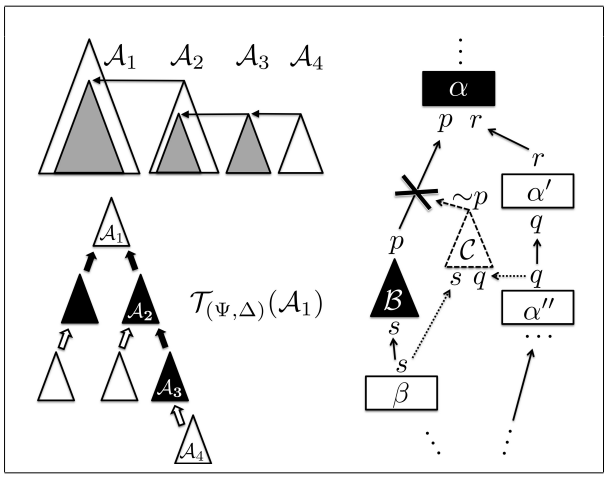


Figure 1. (Top Left) A line of defeaters, with attacked sub-arguments in grey. (Bottom Left) The dialectical tree for $\mathcal{A}_1$, with defeated arguments in black. (Right) A potential (uninteded) argument threat $\mathcal{C}$ for $\mathcal{B}, \alpha$.

A *planning domain* is a tuple $((\Psi, \Delta), A, G)$ for beliefs, actions and goals, representing an agent who wants to find a solution plan: a sequence of actions leading the current Ψ-state into a state where goals G are achieved. As in POP, plans are incrementally built as a set of refinement steps until a (optimal) solution is found. During plan search the planner agent does only impose the minimal constraints on the execution ordering of planned actions, which form a partial order. Actions $\alpha = (\mathsf{P}(\alpha), \mathsf{X}(\alpha), \mathsf{cost}(\alpha))$ have a duration denoted $\|\alpha\| \in \mathbb{N}$, and consist of preconditions $\mathsf{P}(\alpha)$ holding at time $\mathbf{t}(\alpha)$ (a variable), and effects $\mathsf{X}(\alpha)$ holding at $\mathbf{t}(\alpha) + \|\alpha\|$; among these, we assume by default a dummy effect α'ed $\in \mathsf{X}(\alpha)$ stating α *was executed*. The cost $\mathsf{cost}(\alpha)$ is some positive real number. Action α is also denoted $\mathsf{X}(\alpha) \xleftarrow{\alpha} \mathsf{P}(\alpha)$. Ψ and G are encoded as dummy actions $\alpha_{\Psi_t} = (\emptyset, \{\langle \ell, t\rangle \in \Psi\}, 0)$ (for each t) and $\alpha_G = (G, \emptyset, 0)$ with no duration or cost; and related by the constraint: α_{Ψ_0} *occurs before* α_G.

Rules Δ are temporal abstractions of their t-DeLP counterparts: we have now general rules of the form $\delta = \ell \prec (\ell_0, d_0), (\ell_1, d_1), \ldots, (\ell_n, d_n)$, where now (ℓ_n, d_n) denotes the delay (i.e. rule $\langle p, 4\rangle \prec \langle q, 1\rangle, \langle r, 2\rangle$ is an instance of general rule $p \prec (q, 3), (r, 2)$). General rules δ combine into argument steps $\mathcal{A} \subseteq \Delta$, using variables $\langle p_i, \mathbf{t}(\mathcal{A}) + n\rangle$ instead of d_i-values in (p_i, d_i). The duration $\|\mathcal{A}\|$ of $\mathcal{A}$ is defined by the maximum of sums of d_j's in a path of rules from $\mathsf{base}(\mathcal{A})$ to $\mathsf{concl}(\mathcal{A})$. Then n is just $\|\mathcal{A}\|$ minus the sum of the d_js (incl. d_i) in the path of rules from p_i to $\mathsf{concl}(\mathcal{A})$.

A threat can be an argument step, or an argument that is triggered as a side-effect of the plan -as in Figure 1 (Right). The former can only be resolved by reordering the threat $\mathcal{B}$ to the future; for the latter one can also impose arguments defeating the threat.

A plan is a triple $\Pi = (A_\Pi, \mathsf{Goals}(\Pi), \mathbb{I}(\Pi))$ with actions used (non-concurrently), pending goals and inequations expressing constraints on the temporal variables $\mathbf{t}(\kappa)$. The cost of a plan is $\Sigma_{\alpha \in A_\Pi} \mathsf{cost}(\alpha)$. For dummy actions, we may have $\{0 \leq \mathbf{t}(\alpha_G), \mathbf{t}(\alpha_G) \leq 100\} \subseteq \mathbb{I}(\Pi)$, the latter imposing a deadline of 100 time units for any solution plan. Solving a goal in $\mathsf{Goals}(\Pi)$ consists in adding a new constraint to $\mathbb{I}(\Pi)$: imposing the plan step κ before the step κ' whose $\mathsf{base}(\cdot)$ or $\mathsf{P}(\cdot)$ set contain the literal being solved:

add $\mathbf{t}(\kappa) + \|\kappa\| \leq \mathbf{t}(\kappa')$. Moving a threat to the future, κ' being the step supported by the threatened step). A non-deterministic search algorithm for the space of plans consists in:

0. Start with the empty plan $\Pi_\emptyset = (\emptyset, G, \mathbb{I}(\Pi_\emptyset))$.
1. If an unsolved goal or threat exists, choose a goal or threat. Otherwise terminate.
2. Choose some action- or argument-step, or resp. some threat resolution move (if Π cannot be refined, then backtrack to the parent node). Refine Π.
3. Update the set of unsolved goals and threats. Go to step 1.

Given a solution plan Π for $((\Psi, \Delta), A, G)$, any sequential execution of A_Π, given by a model $\tau : A_\Pi \to \mathbb{N}$ of the inequations $\mathbb{I}(\Pi)$ will enforce G according to t-de.l.p. defined by Ψ-plus-actions' effects $\mathsf{X}(\tau(\alpha))$, for any $\alpha \in \Pi$ (and conditional on their preconditions being warranted). The results are stronger if $\mathbf{t}(\alpha_G)$ is imposed a bound.

Theorem 2 *[9] The search algorithm is correct; under temporal deadlines $\mathbf{t}(\alpha_G) \leq k$, the algorithm is complete. Uniform cost heuristic for A^*-search is admissible.*

3. Multi-agent issues in t-DeLP-POP for ABN

Understanding a plan. Let $\mathsf{Ag} = \{1, \ldots, n\}$ be a set of agents. Each agent i is initially endowed with a planning domain $\mathbb{M}_i = ((\Psi_i, \Delta_i), A_i, G_i)$. $\mathsf{Plans}(\mathbb{M})$ will denote the plans according to domain $\mathbb{M}$. For an agent j to understand a plan Π for some $\mathbb{M}$ (e.g. $\mathbb{M} = \mathbb{M}_i$) communicated by i, it is enough the components of Π are in $\mathbb{M}_j$: literals at Ψ supporting the plan are in Ψ_j; Δ rules used for argument steps are in Δ_j, $A_\Pi \subseteq A_j$[6].

Relativity of threats w.r.t. beliefs. A plan $\Pi = (A_\Pi, \mathsf{Goals}(\Pi), \mathbb{I}(\Pi))$ is defined without threats, because these are relative to the t-de.l.p. (Ψ_i, Δ_i) of the agent i evaluating this plan. For i to detect a threat $\mathcal{B}$ it suffices that $\mathcal{B} \subseteq \Delta_i$. Put the other way round, one can understand a plan without agreeing on the set of threats existing (i.e. on whether Π is a solution for G). Hence, the utility of or preference for a plan can also change due to learning. The set of threats in Π according to $\mathbb{M}$ will be denoted $\mathsf{Threats}^{\mathbb{M}}(\Pi)$.

Commications as expansions of $\mathbb{M}_i$. A dialogue turn will consist in agent i communicating an offer, argument, etc. to an agent j. The information contained (facts, rules, actions) is extracted and learnt by j, by expanding $\mathbb{M}_j$, resp., expanding Ψ_j, Δ_j, A_j (goals are also affected; see offers below). Modeling other agents is done by (instrumental) planning domains of the form $\mathbb{M}_{ji}$, used by j to reason (and exploit) i's difficulties solving G_i (from the point of view of j's own know-how). Since agents' planning domains vary with time, $\mathbb{M}_i$ will be added a superscript denoting the expanded domain at some turn.

Negotiation roles; concession functions. An agent j's service to enforce ℓ, for some customer i's plan Π_i, is an action $\alpha_j \in A_j$ such that $\ell \in \mathsf{X}(\alpha_i) \cap (\mathsf{Goals}(\Pi_i) \cup \overline{\mathsf{Threats}^{\mathbb{M}_i}(\Pi_i)})$, or an argument $\mathcal{A}$ with $\mathsf{concl}(\mathcal{A}) = \ell$ (the latter at null cost) for similar ℓ. Within a dialogue taking place between agents i and j unfolds, a sub-dialogue between j and k may be triggered ultimately motivated by i's goals.

Since our focus here is in the argumentation aspects of ABN, we will just assume agents are endowed with concession functions: the input of agent i's concession function

[6] An action, say, $\alpha_i \in A_j$ with $i \neq j$ is merely informative. An offer received, or an agreement about α_i, say an exchange of α_i for β_j, will be represented as a new action $(\alpha_i \otimes \beta_j)_j$ *of agent* j in A_j.

w.r.t. an agent j is the set of rival offers addressing the same goal. The output is a new offer improving rival offers that, resp., w.r.t. j's interests, or i's interests.

We might assume as well each agent has a *communication policy* that regulates when one's information (relevant in the dialogue) is not sent, to preserve one's interests. In this paper, though, we assume agents are honest, so all relevant information will be sent.

Social Relations. Finally, we note how social relations between negotiating agents may alter the development of a negotiation dialogue. Among basic social relations in $\mathsf{Ag} \times \mathsf{Ag}$, we consider: cooperative $\equiv_{\mathsf{co}}$, equitable $\simeq_{\mathsf{eq}}$ and hierarchical $\prec_{\mathsf{hi}}$. We simplify by assuming these to be, resp., equivalence relations ($\equiv_{\mathsf{co}}$, $\simeq_{\mathsf{eq}}$) and a partial order $\prec_{\mathsf{hi}}$ in Ag. Agents $a \equiv_{\mathsf{co}} b \equiv_{\mathsf{co}} \ldots$ are cooperative if they share goals $G_a = G_b = \ldots$ (i.e. they adopt each other goals, if consistent); a unique groupwise dialogue suffices for $\equiv_{\mathsf{co}}$ (see [10]). Equitable agents $a \simeq_{\mathsf{eq}} b$ are free to withdraw from a negotiation and need not justify the absence of offers. In contrast, within a hierarchy, say with agents $b \prec_{\mathsf{hi}} a$, the power of a consists in a set of (tacit) agent threats to b, which (by law) cannot be replied or counter-argued. This relation $\prec_{\mathsf{hi}}$ demands a new speech act, b *challenging* a, that consists showing the demands cannot be met, by disclosing to a one's actions/knowledge. (To see this, a must fail to find a plan using this information.)

Speech acts. We define next the speech acts listed in Section 1. These divide into: proposals (offers, demands, acceptance), theoretical arguments (plan threats, agent threats) and practical arguments (persuasion, challenge):

Offers	I propose that I do α if you do β
Demands	I have goal g in plan Π; can you help?
Plan threats	Your/his offer conflicts with this part of my/your plan
Agent threats	If you permit/cause ℓ to occur, I swear I will do α
Persuasion	This agent offers me so-and-so, can you match this offer?
Challenge	In this plan, your demands cannot be met under such deadline, or at all

Argument steps for an agent i's plan Π_i or argument threats to Π_i are generated as in the single-agent case but for Π as a plan in the "planning domain" $(\Psi_j, \Delta_j, A_j, G_i)$.

Definition 1 *Let $\{x, y\} \subseteq \mathsf{Ag}$, with plans Π_x, Π_y. An* offer *κ from x to y is:*

- *an arg. step $\kappa = \mathcal{A} \subseteq \Delta_x$. This updates $\mathbb{M}_y$ as: $\mathbb{M}_y = ((\Psi_y, \Delta_y \cup \mathcal{A}), \ldots)$.*
- *some service exchange $\kappa = (\alpha_x \otimes \alpha_y, \mathbb{I}(\alpha_x), \mathbb{I}(\alpha_y))$. Read the old $\mathbb{M}_y$ as: "the offer is rejected by y"; and model its acceptance in a new $\mathbb{M}'_y = (\ldots, A_y \cup \{\kappa\}, G_y \cup \{\alpha_y\text{'}\mathsf{ed}\})$, where $(\alpha_x \otimes \alpha_y) = (\emptyset, \mathsf{X}(\alpha_x), 0)$.*
- *some purchase offer $\kappa = (\alpha_x \otimes n, \mathbb{I}(\alpha_x))$. We just expand $A_y \in \mathbb{M}_j$ with $\kappa = (\alpha_x \otimes n) = (\emptyset, \mathsf{X}(\alpha_x), n)$; (and similarly for $\mathbb{M}_{x,y}$). Or,*
- *some purchase offer $(\alpha_y \otimes n, \mathbb{I}(\alpha_y))$. Create $\mathbb{M}'_y = (\ldots, A_y \cup \{(\alpha_y \otimes n)\}, G_y \cup \{\alpha_y\text{'}\mathsf{ed}\})$ as above, with $(\alpha_y \otimes n) = (\mathsf{P}(\alpha_y), \mathsf{X}(\alpha_y), \mathsf{cost}(\alpha_y) - n)$.*

where $\mathbb{I}(\alpha_x)$ (resp. $\mathbb{I}(\alpha_y)$) is a set of constraints $\mathbf{t}(\alpha_x) \leq / \geq m$ derived from those for action α_x in $\mathbb{I}(\Pi_x(\alpha_x \otimes \alpha_y)$ (resp. for action $(\alpha_x \otimes \alpha_y)$). Here $m = \Sigma_{\mathbf{t}(\kappa) \leq \mathbf{t}(\kappa') \leq \mathbf{t}(\alpha_{G_x})} \|\kappa'\|$. All changes to $\mathbb{M}_y$ in this definition apply to $\mathbb{M}_{xy}$ as well.

Thus, asking x to add a new constraint $\mathbf{t}(\alpha_x)$ to a previous offer $(\alpha_x \otimes n)$ is a new negotiation and can end up in a different agreement $(\alpha_x \otimes n')$, reflecting x's opportunity cost for α_x under the new temporal constraints $\mathbb{I}(\Pi_x) \leftarrow \mathbb{I}(\Pi_x) \cup \mathbb{I}(\alpha_x)$.

Offers to be sent are generated according to one's *concession policy*. Similarly for threats (see below) which aim to modify the consequences of rival plans, and hence the agent's evaluation of these plans (i.e. her preference relation among plans).

Definition 2 *Let agents $i, j \in \mathsf{Ag}$ with $\mathsf{Goals}(\Pi_i) \neq \emptyset$, for some Π_i current plan in $\mathsf{Plans}(\mathbb{M}_i)$). Components of the planning domain $\mathbb{M}_x$ are expanded as follows:*

demand$_{i \triangleright j}$*:* Π_i	*(j)* $\Pi_i \in \mathsf{Plans}(\mathbb{M}_{ji})$
offer$_{j \triangleright i}$*:* $\Pi_i(\kappa)$	*(i)* $\kappa \subseteq \Delta_i$ *or* $(\alpha_i \otimes \beta_j) \in A_i$ *(similarly for* $\mathbb{M}_{ji}$*)*
plan threat$_{j \triangleright i}$*:* $\mathcal{B}$	*(i)* $\Delta_i \cup \mathcal{B}$
agent threat$_{j \triangleright i}$*:* (ℓ, α)	*(i)* $\Delta_i \cup \{\ell' \prec (\ell, \|\alpha\|) \mid \ell' \in \mathsf{X}(\alpha)\}$.
	(j) $G_j \cup \{p^\star\}$*, for the new literal* $p^\star :=$ satisfied-(ℓ, α) *, with*
	(j) $\{p^\star \prec \sim(\ell, 0); p^\star \prec (\ell, 0), \mathsf{X}(\alpha) \times \{0\}\} \subseteq \Delta_j$
persuasion$_{i \triangleright j}$*:* $\Pi_i(\kappa_k)$	*(j)* $\Pi_i(\kappa_k) \in \mathsf{Plans}(\mathbb{M}_{ji})$.
challenge$_{j \triangleright i}$*:* Π_i	$\mathbb{M}_j = (\Psi_j \cup \Psi_i^*, \Delta_j \cup \Delta_i^*, A_j \cup A_i^*, G_j)$.

where the relevant subsets $\Psi_i^*, \Delta_i^*, A_i^*$ (of $\mathbb{M}_i$) are defined as in the dialogues of [10].

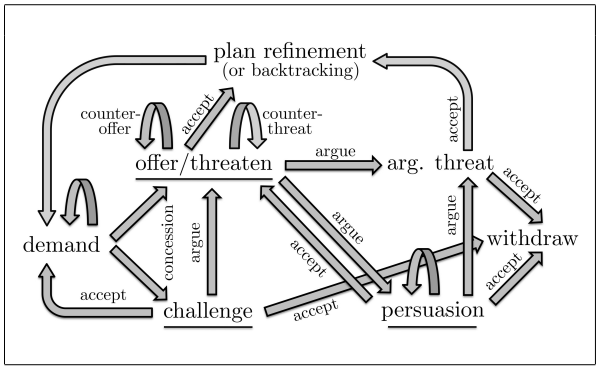


Figure 2. The ABN protocol. Underlined moves may involve nested dialogues with new agents.

Note that a threatened agent i will make public the threats received by j to the other parties to point out these new existing "argument threats" in their suggested plans.

Practical preference. During plan search two notions of preference play a role. For plan refinement, the best plan is selected by each agent i, solely based on its cost $\mathsf{cost}(\Pi) := \Sigma_{\alpha \in A_\Pi} \mathsf{cost}(\alpha)$. The remaining plans can still be "improved" by persuading other agents by showing the existence of better rival offers. In this persuasion moves, the preference criterion has stronger conditions, namely, $\Pi \succeq_i^{\mathbb{M}_{ji}} \Pi'$ if and only if

$$\mathsf{Threats}^{\mathbb{M}_{ji}}(\Pi) \subseteq \mathsf{Threats}^{\mathbb{M}_{ji}}(\Pi'), \mathsf{Goals}(\Pi) \subseteq \mathsf{Goals}(\Pi') \text{ and } \mathsf{cost}(\Pi) \leq \mathsf{cost}(\Pi)$$

with one of these inequalities being strict: resp., $\subsetneq$ or $\subsetneq$ or $<$. Here $\mathbb{M}_{ji}$ denotes $\mathbb{M}_i$ if $j = i$. Note this preference criterion is subject to discussion because of the relativity of threats. Hence, a persuasion move can be replied by a plan threat, which can be replied by a persuasion move from a refined plan, etc. (see Figure 2).

The preference $\succeq_i^{\mathbb{M}_{ji}}$ restricts j's counter-offers to i: given two rival plans $\Pi_i \succeq_i^{\mathbb{M}_{ji}} \Pi_i'$, involving offers with different agents and Π_i' containing an offer $(\alpha_i \otimes \beta_j)$, new offers from j, say $(\alpha_i' \otimes \beta_j')$, must make the new plan Π_i'' improve Π_i': $\Pi_i'' \succeq_i^{\mathbb{M}_{ji}} \Pi_i'$. In the particular case where $\Pi_i = \Pi_i^\star(\ldots)$ and $\Pi_i' = \Pi^\star(\alpha_i \otimes \beta_j)$, new offers Π_i'' must be competitive: $\Pi_i'' \succeq_i^{\mathbb{M}_{ji}} \Pi_i$.

4. A protocol for ABN dialogues.

Even if dialogues are assumed to occur concurrently, we model them as taking place in a sequential way. Thus, turns n encode a pair $f : n \mapsto \langle i, j\rangle$ of agents i and j, resp. the sender at and the receiver of information at turn n; here $f : \mathbb{N} \to \mathsf{Ag}^2$ is an enumeration of Ag^2 always following the same order, i.e. satisfying $f(n) = f(n + |\mathsf{Ag}| \cdot (|\mathsf{Ag}| - 1))$. The content sent at a turn of the form $\langle i, j\rangle$ is fully determined by the current $\mathbb{M}_i$, $\mathbb{M}_{ij}$ domains, and previous (rival) offers in turns of the form $\langle i, j\rangle$ or $\langle k, i\rangle$ (where $k \neq i, j$).

Definition 3 *Given turn $n \xrightarrow{f} \langle i, j\rangle$, the messages sent by i, and the changes in j are:*

- *Each message of i at n follows the next-step(s) in Figure 2, for each message in the previous turn of the form $\langle j, i\rangle$. Agent i updates $\mathbb{M}_{ij}$, similarly to that of $\mathbb{M}_j$, see next.*
- *The components of each domain $\mathbb{M}_j, \mathbb{M}_j', etc$ are updated by each message received by i as usual. Agent j updates each set $\mathsf{Plans}(\mathbb{M}_j), \mathsf{Plans}(\mathbb{M}_j')$, etc. with new refine–ment steps available, as well as (re-)evaluating $\mathsf{Threats}^{\mathbb{M}_j}(\Pi)$, etc. for each (explored) plan Π in old $\mathsf{Plans}(\mathbb{M}_j)$, etc. A new $\mathbb{M}_j'$ is created for each new offer received.*

Proposition 1 *Given a sequence $(\mathbb{M}_i)_{i\in\mathsf{Ag}}$, and concession functions γ_i, the ABN dialogues terminate in a finite time.*

(Proof Sketch) The reason is that: since $\mathsf{Ag}, \mathsf{Var}$ are finite and each $\mathbf{t}(\alpha_{G_x})$ is bounded, each possible set Ψ_x, Δ_x, A_x and G_x is finite, and so is the number of $\mathbb{M}_x$'s for each $x \in \mathsf{Ag}$. Also, the set of offers is finite. Hence, the refinements of any Π must be finite, and the length of $\Pi \in \mathsf{Plans}(\mathbb{M})$ must be bounded. Thus, each set $\mathsf{Plans}(\mathbb{M}_x)$ is finite. Thus, the negotiation space is finite and so is any dialogue between x and y.

Once a dialogue terminates, which agreements are reached depends on whether we take offers $(\alpha_i \otimes \beta_j)$ sent by i as committing i (if accepted by j) or not. If they do, the previous ABN protocol determines which agreements are reached. If offers do not commit agents that proposed them, the agreement problem turns out to be that of forming overlapping coalitions in the line of [1] (i.e. a problem in cooperative game-theory).

4.1. Case study: negotiating a deadline.

In the following, we assume action $\{\mathsf{have_x(o)}, \sim\mathsf{have_y(o)}\} \xleftarrow{\mathsf{give_{y\triangleright x}(o)}} \{\mathsf{have_y(o)}\}$ is known by all agents.

Example 1 *Given a company with manager* a *and a vetting agent* b *such that* $\mathsf{b} \prec_{\mathsf{hi}} \mathsf{a}$, *let* $\mathsf{c_0}$ *be a customer asking* a *to change their current deal. (1)* a *orders (i.e. demands)* b *to vet all the customers* $\mathsf{c_0}, \ldots, \mathsf{c_n}$ *in the area, in 8 hrs. (2)* b *argues this cannot be done (i.e.* b *challenges* a*). As a reply, the manager* a *can send two new demands: (3) to have*

just the important customer c_0 *vetted; or (3') to keep the goals but extend the deadline to tomorrow morning, i.e. in 24 hrs. Agent* b *sends solution plans, resp., at lines (4) and (4'). In (4')* b *replies by asking to be paid for the extra work (at one coin per extra hour).*

---

(1) $demand_{\mathsf{a}\triangleright\mathsf{b}}$	$\Pi_{\mathsf{a}} = (\emptyset, \{\mathsf{have}_{\mathsf{a}}(\mathsf{docu}(\mathsf{c}_{\mathsf{k}}))\}_{\mathsf{k}\leq\mathsf{n}}, \{\mathbf{t}(\alpha_{\mathsf{G}_{\mathsf{a}}}) \leq 8\})$
(2) $challenge_{\mathsf{b}\triangleright\mathsf{a}}$	$A^*_{\mathsf{b}} = \{\mathsf{have}_{\mathsf{b}}(\mathsf{docu}(\mathsf{c}_{\mathsf{k}})\} \xleftarrow{\mathsf{vet}_{\mathsf{b}}(\mathsf{c}_{\mathsf{k}})}\}$, with $\|\cdot\| = 3\}$
	$\Psi^*_{\mathsf{b}} = \{\sim\mathsf{have}_{\mathsf{b}}(\mathsf{docu}(\mathsf{c}_{\mathsf{k}}))\}_{\mathsf{k}\leq\mathsf{n}}\}$
	a *checks* $\mathbb{M}^*_{\mathsf{a}} = ((\Psi_a \cup \Psi^*_b, \Delta_a), A_a \cup A^*_b, G_a)$ *has no solution.*
(3) $demand_{\mathsf{a}\triangleright\mathsf{b}}$	$\Pi_{\mathsf{a}} = (\emptyset, \{\mathsf{have}_{\mathsf{j}}(\mathsf{docu}(\mathsf{c}_0))\}, \{\mathbf{t}(\alpha_{\mathsf{G}_{\mathsf{a}}}) \leq 8\})$
(4) $offer_{b\triangleright a}$	$\Pi_{\mathsf{a}}(\{\mathsf{give}_{\mathsf{b}\triangleright\mathsf{a}}(\mathsf{docu}(\mathsf{c}_0))), \mathsf{vet}_{\mathsf{b}}(\mathsf{c}_0)\}, \emptyset, \{\mathbf{t}(\mathsf{vet}_{\mathsf{b}}(\mathsf{c}_0)) + 3 < 8\})$
(3') $demand_{\mathsf{a}\triangleright\mathsf{b}}$	$\Pi_{\mathsf{a}} = (\emptyset, \{\mathsf{have}_{\mathsf{a}}(\mathsf{docu}(\mathsf{c}_{\mathsf{k}}))\}_{\mathsf{k}\leq\mathsf{n}}, \{\mathbf{t}(\alpha_{\mathsf{G}_{\mathsf{a}}}) \leq 24\})$
(4') $offer_{\mathsf{b}\triangleright\mathsf{a}}$	$\Pi_{\mathsf{a}} = (\{(\mathsf{give}_{\mathsf{b}\triangleright\mathsf{a}}(\mathsf{docu}(\mathsf{c}_{\mathsf{k}})) \otimes 3), \mathsf{vet}_{\mathsf{b}}(\mathsf{c}_{\mathsf{k}})\}_{\mathsf{k}\leq\mathsf{n}}, \emptyset,$
	$\{\mathbf{t}(\mathsf{vet}_{\mathsf{b}}(\mathsf{c}_{\mathsf{k}})) + 3 \leq \mathbf{t}(\mathsf{vet}_{\mathsf{b}}(\mathsf{c}_{\mathsf{k}+1})) \leq 24\}_{\mathsf{k}\leq\mathsf{n}})$

4.2. Case study: hanging picture and mirror.

Example 2 *(Adapted from [14]) Agent* a*, endowed with a nail and a picture, wants to hang the picture:* $\mathsf{picture.in.wall}^{\mathsf{a}} \in G_{\mathsf{a}}$*; while* b*, endowed with a hammer and a mirror, wants to hang the mirror while keeping the hammer:* $\mathsf{mirror.in.wall}^{\mathsf{b}}, \mathsf{have}_{\mathsf{b}}(\mathsf{hammer}) \in G_{\mathsf{b}}$*. Both agents' deadlines are set to 4 hrs. Agents' actions* A_{x} *are defined for any* $\mathsf{x}, \mathsf{u} \in \{\mathsf{a}, \mathsf{b}\}$*, where* x *denotes the executing agent, and* u *the proprietor of* $\mathsf{wall}^{\mathsf{u}}$*:*

$$\{\mathsf{nail.in.wall}^{\mathsf{u}}_{\mathsf{x}}\} \xleftarrow{\mathsf{hammer}(\mathsf{nail.wall}^{\mathsf{u}})_{\mathsf{x}}} \{\mathsf{have}_{\mathsf{x}}(\mathsf{hammer}), \mathsf{have}_{\mathsf{x}}(\mathsf{nail})\}$$
$$\{\mathsf{picture.in.wall}^{\mathsf{u}}_{\mathsf{x}}\} \xleftarrow{\mathsf{hang}(\mathsf{picture.wall}^{\mathsf{u}})_{\mathsf{x}}} \{\mathsf{nail.in.wall}^{\mathsf{u}}, \mathsf{have}_{\mathsf{x}}(\mathsf{picture})\}$$
$$\{\mathsf{mirror.in.wall}^{\mathsf{u}}_{\mathsf{x}}\} \xleftarrow{\mathsf{hang}(\mathsf{mirror}^{\mathsf{u}})_{\mathsf{x}}} \{\mathsf{nail.in.wall}^{\mathsf{u}}, \mathsf{have}_{\mathsf{x}}(\mathsf{mirror})\}$$

No agent's goal g *is assumed to persist* $\delta_g \notin \Delta_{\mathsf{u}}$*, since persistence of* g *is dependent on that of* $\mathsf{nail.in.wall}^{\mathsf{u}}$*. In fact,* b *is sick, so she cannot hammer strong enough. This is modeled by rules in* $\Delta_{\mathsf{a}}, \Delta_{\mathsf{b}}$*:* $\langle \mathsf{weak.nail.wall}^{\mathsf{u}}_{\mathsf{b}}, \rangle \prec \langle \mathsf{nail.in.wall}^{\mathsf{u}}_{\mathsf{b}}, \mathsf{t}\rangle, \langle \mathsf{sick}_{\mathsf{b}}, \mathsf{t}\rangle$ *and* $\langle \sim\mathsf{nail.in.wall}^{\mathsf{u}}, \mathsf{t}+2\rangle \prec \langle \mathsf{nail.in.wall}^{\mathsf{u}}, \mathsf{weak.nail.in.wall}^{\mathsf{u}}_{\mathsf{b}}, \mathsf{t}\rangle, \langle \mathsf{weak.nail.wall}^{\mathsf{u}}, \mathsf{t}\rangle$.

After initial demands, a *offers* $\mathsf{give}_{\mathsf{a}\triangleright\mathsf{b}}(\mathsf{nail}) \otimes (\mathsf{give}_{\mathsf{b}\triangleright\mathsf{a}}(\mathsf{hammer}) \otimes \mathsf{give}_{\mathsf{a}\triangleright\mathsf{b}}(\mathsf{hammer}))$ *with obvious constraints. Then* b *challenges the resulting plan by communicating he is sick; agent* a *is able to detect a threat in the proposed plan to* b*. Then,* b *proposes a new offer* $\mathsf{hammer}(\mathsf{nail.wall}^{\mathsf{b}})_{\mathsf{a}} \otimes (\mathsf{give}_{\mathsf{b}\triangleright\mathsf{a}}(\mathsf{hammer}) \otimes \mathsf{give}_{\mathsf{a}\triangleright\mathsf{b}}(\mathsf{hammer}))$.

4.3. Case study: threats in a 2*-vendors* 1*-customer scenario.*

Example 3 *The manager* a *of a building company learns a container of bricks has been stolen from a building. She phones (among others) agent* b *from a security company, who offers to protect the building for* m *coins per night (the action* $\mathsf{protect}_{\mathsf{c}}$ *at* t *does preserve all* $\langle \mathsf{have}_{\mathsf{a}}(\cdot)$ *literals up to hour* $t + 8$*). Agent* a *replies by producing a better offer at cost* m' *from another agent* c*. Then,* b *suggests that some robberies could affect* again *the construction schedule for this building. The rule* $\langle \sim\mathsf{have}_{\mathsf{b}}(\mathsf{bricks}), 24\rangle \prec \langle \sim\mathsf{protect}_{\mathsf{c}}\text{'ed}, 0\rangle, \langle \mathsf{protect}_{\mathsf{b}}\text{'ed}, 0\rangle$ *is learnt by the manager. Since* $\mathsf{cost}(\mathsf{buy}(\mathsf{bricks})) + \mathsf{m}' > \mathsf{m}$*, agent* a *may decide to pay* m *to* b*. Or, if a po-*

liceman p *exists,* a *can reply by (counter-) threatening* b *to inform* p*. Since the rule* $\mathsf{jailed_b} \multimap (\mathsf{inform_{a \triangleright p}(criminal(b))}, 4)$ *is in both sets* Δ_{ab} *and* Δ_{b}*, agent* c *cancels her own threat, so* b *deletes the corresponding rule, and* c *withdraws from the dialogue.*

5. Conclusions and Future work.

We have presented a protocol for negotiation dialogues that take place within a society of planner-reasoner agents. We showed some basic properties, and studied the dialogues for several ABN scenarios extracted from the literature. These case studies show the proposed framework for the agent architecture is quite expressive for ABN scenarios. As for future work, we would like to study the relation between algorithmic search and game-theoretic properties of agreements in the proposed ABN protocol: e.g. Pareto-optimality of the sequence of plans agreed upon, etc.

References

[1] M. Albizuri, J. Aurrecoechea, J. Zarzuelo *Configuration values: Extensions of the coalitional Owen value*, Games and Economic Behavior 57, pp. 1–17 (2006)

[2] L. Amgoud, Y. Dimmopoulos, and P. MoraitisÊ*A Unified and General Framework for Argumentation-based Negotiation* Proc. of AAMAS 2007

[3] L. Amgoud and H. Prade *Formal Handling of Threats and Rewards in a Negotiation Dialogue*, Proc of ArgMAS 2005

[4] A. Artikis, M. Sergot and J. Pitt *An executable specification of a formal argumentation protocol* Artificial Intelligence 171, pp. 776Ð804 (2007)

[5] Y. Dimopoulos, P. Moraitis, L. Amgoud. *Characterizing the Outcomes of Argumentation-Based Integrative Negotiation*, Proc. of IAT 2008, 456-460.

[6] A. García and G. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4:95–138, 2004.

[7] D. García, A. García, and G. Simari. Defeasible reasoning and partial order planning. In Proc. of the *5th International Conference on Foundations of information and knowledge systems, FoIKS 2008*, LNCS 4932, pp. 311–328, 2008.

[8] J. Grant, S. Kraus and D. Perlis *A logic-based model of intention formation and action for multi-agent subcontracting*, Artificial Intelligence, 163. 163–201, 2005.

[9] P. Pardo *Temporal defeasible logic-based planning t-DeLP-POP for physically correct plans.* Available at http://www.iiia.csic.es/files/pdfs/Pardo-GenPlanAAAI11.pdf.

[10] P. Pardo. S. Pajares, E. Onaindía, L. Godo, P. Dellunde. *Multi-agent argumentation for cooperative planning in DeLP-POP*, Proc. of AAMAS 2011, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2-6, 2011, Taipei, Taiwan, pp. 971-978.

[11] P. Pardo and Lluís Godo *t-DeLP: a temporal extension of the defeasible logic programming argumentative framework* (Submitted). Available at http://www.iiia.csic.es/files/pdfs/tDeLP-submitted.pdf

[12] J. Pearl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley, 1984.

[13] J. Penberthy and D. Weld. Ucpop: A sound, complete, partial order planner for adl. In Proc. of the *3rd International Conference on Knowledge Representation and Reasoning (KR'92)*, pp. 103–114, 1992.

[14] S. Parsons, C. Sierra and N. Jennings *Agents that reason and negotiate by arguing*. Journal of Logic and Computation 8 (3), 261-292, 1998.

[15] C. Sierra, N. Jennings, P. Noriega and S. Parsons *A framework for argumentation-based negotiation*, In Proceedings of the 4th International Workshop on Intelligent Agents IV, Agent Theories, Architectures, and Languages (ATAL'97), 177 - 192, 1997.

[16] K. Sycara, S. Kraus and A. Evechnik *Reaching agreements through Argumentation: A Logical Model and Implementation*, Artificial Intelligence, 104 (1-2), pp. 1-69 (1998)

Biologically Inspired Turn Control for Autonomous Mobile Robots

Xavier PEREZ-SALA [a], Cecilio ANGULO [a] and Sergio ESCALERA [b]

[a] *CETpD-UPC. Technical Research Centre for Dependency Care and Autonomous Living, Universitat Politècnica de Catalunya, Neàpolis, Rambla de l'Exposició, 59-69, 08800 Vilanova i la Geltrú, Spain*

[b] *MAiA-UB. Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, 08007 Barcelona, Spain*

Abstract. An exportable and robust system for turn control using only camera images is proposed for path execution in robot navigation. Robot motion information is extracted in the form of optical flow from SURF robust descriptors of consecutive frames in the image sequence. This information is used to compute the instantaneous rotation angle. Finally, control loop is closed correcting robot displacements when it is requested for a turn command. The proposed system has been successfully tested on the four-legged Sony Aibo robot.

Keywords. Robot navigation, Path execution, Human motion, Turn control, SURF descriptor

Introduction

Navigation for autonomous mobile robots, independently of the platform and its task, implies to solve two related problems: path planning and path execution. Path planning can be defined as a high level robot guidance from a place to another place or from one orientation to another one, while path execution refers to low level processes needed to fulfill path planning decisions [6]. This work is about, given a certain path plan, how to ensure a successful turn control in path execution when the only available information for the robot is data extracted from its on-board camera. Remarkably, no landmarks in the environment are needed.

Unexpected robot behaviours can be observed during path execution when a system is asked for reaching a place or set point, though it acted properly in simulated or ideal conditions. Failures in path execution, even for simple path executions like a 'go straight forward' or 'turn 23^o'path commands, are due to several reasons: noise in the sensors, damages in the actuators, perturbations, model errors or collisions. Consequently, a feedback control would be interesting to be implemented to correct the robot from possible motion deviations.

A common approach for obtaining feedback is to consider artificial landmarks [5,6]. However, for a general solution, no landmark should be considered. Another solutions focus on constraining robot motion and camera localization on the robot in order to ob-

tain robot egomotion [2,3,4]. Since nor robot configuration, neither camera localization will be constrained, but be placed in the front direction, egomotion can not be considered.

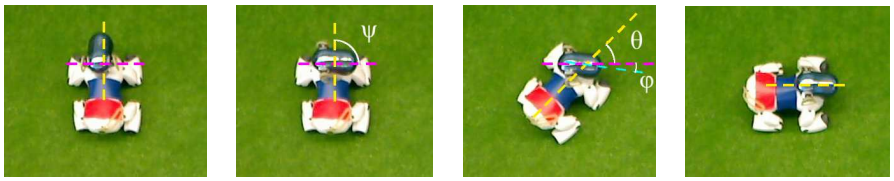
The general problem at hands is to ensure the execution of turning a certain angle by a general mobile robot endowed with a rotating camera, when the only available information are frames from the camera and angle measurement of the rotation parallel to the ground, between the robot and the camera. Examples of robot configurations which could use this algorithm are: biological inspired robots with "head" (mobile part of the robot where camera is placed) and "body" (the rest of the robot) joined by a motorized neck with encoders, or common robots with an embedded camera with pan degree of freedom. Our proposed approach starts rotating the head to the desired angle using its sensor. Then, the body of the robot is aligned with the head, maintaining its orientation. In order to maintain the focus of the head in the same direction, the robot rotation is computed and compensated. Similarly to other approaches based on optical flow [2], consecutive frames are used to extract an approximation of the robot movement, by observing 2-D displacements of brightness patterns in the image. However, unlike standard solutions, the robot rotation will be computed online by extracting Speeded Up Robust Features (SURF) from image key-points and computing its displacement, i.e. motion information from SURF robust descriptors of consecutive frames of image sequences provided by the robot camera. Optical flow is a measure closely related with motion field [1], i.e. the projection of 3-D relative velocity vectors of the scene points onto the 2-D image plane. During a rotation, motion field shows almost parallel vectors with the same length, closely related to the rotation angle. It is proposed in this work to achieve control of turn for mobile robots by computing rotation angle from the vectors of the SURF flow. This knowledge will be the only information needed to close the control loop, and to achieve the desired rotation.

The rest of the paper is organized as follows: next section overviews the state-of-the-art methods for robot navigation based on optical flow. Section 2 describes the solution proposed for robot rotation. In Section 3, experiments are presented and discussed. Finally, conclusions and further research lines are listed in Section 4.

1. Background

Visual based solutions for autonomous robot navigation are typically focused on path planning or path execution through localization computation. Whether it is possible to set-up the environment, standard approaches consist on the use of artificial landmarks to provide an accurate localization [5,6]. Otherwise, if some restrictions can be taken over the robot configuration (i.e. camera position or robot movement), solutions focus on egomotion computation [2,3,4] in order to fulfill a localization based on visual odometry. However, if nor environment is adaptable neither restrictions are taken, for a general solution it is proposed to keep away from previous approaches, mimicking human motion.

Human motion suffers a rapid evolution in childhood ages, during the period when children learn to walk like adults. Similarly to path execution in robots, goal-oriented locomotion in humans implies three abilities: localizing the visual target, controlling locomotor performance, and appropriately organizing visual-motor interface [10]. In early ages, spatial localization is achieved with respect to the child's own body position. In the



(a) (b) (c) (d)

Figure 1. 90^o rotation process: From any initial stage (a), robot camera is rotated in the desired angle ψ (b). Then, the body rotates in the direction where the head is pointing while the camera maintains the same orientation (c) until body and camera are aligned (d).

next stage, egocentric representation of the environment is abandoned while children use temporal landmarks present in the environment to organize the movements and positions, i.e. in order to fulfill intermediate goals in a path. Children finally walk like adult humans when become capable of building reliable exocentric topographic representations. Moreover, during the first years of independent walking, the head is progressively stabilized relative to space, facilitating the interpretation of the environment during locomotion. In addition, anticipatory strategies emerge to orient the head movements during locomotion tasks [10,11]. Anticipatory movements of eyes and head in the direction of the trajectory is essential in obstacles avoidance and, also, in following path constraints [12].

Simulating human motion, in a previous work it was presented a navigation control where, using the ideas of qualitative egocentric motion, it is performed a control to follow straight forward paths [14]. Visual features were extracted from robot camera through SURF flow, and used as temporal landmarks centered to respect the camera reference frame [13]. Inspired by human motion, a novel approach is presented here to control turns in robots with a rotating camera. Using camera as a "head", it anticipates turns in the direction the robot intends to go, facilitating the interpretation of the environment during locomotion. Moreover, SURF flow is also considered to compute the rotation angle and close the loop, controlling the robot turn.

2. Turn Control in Robot Navigation

A method to control turns during the navigation of mobile robots is introduced. A closed loop is implemented to control the robot turn, with feedback signal extracted from on-board camera images. The proposed procedure (Fig. 1) is composed by three steps: firstly, the head is rotated in the desired angle using its encoder, i.e. set point of the control is fixed. Next, in the *body alignment* step, through the use of SURF flow robot starts to rotate in the direction the head is pointing while camera is maintained in the same orientation. Finally, it is checked that turn is completed when body and head are completely aligned. At the same time, body-head alignment is composed by two simultaneous movements: body and head controls. Body control is responsible for rotating the robot, depending on the difference between body and head angles. Head control consists on maintaining the same head orientation during all the process, rotating the head in the same, unknown, angle but in the opposite direction.

2.1. Feedback Control

From any initial stage (Fig. 1(a)), the head is rotated to the desired angle ψ (Fig. 1(b)). In order to fulfill this step, a position control is done using the neck encoder [1]. Given this robot configuration, the process for body alignment rotates the body in the direction that the head was turned (see body turn in image sequence Fig. 1(b) - Fig. 1(d)), while head orientation is held during the same sequence. Hence, the alignment process is composed by two movements which have their own feedback control: body control and head control.

Body control searches for aligning the neck through rotating the whole robot, while head control regulates the head orientation using external references, balancing the robot turns by rotating the head in the opposite direction with the same, unknown, angle. Thus, the set point for body control is to recover head-body alignment, with error signal θ being the angle between body and head, provided directly by the robot sensor: "pan" angle. Actuation is applied on the rotation velocity of the robot, since sensors are not used to define the robot orientation. On the other hand, head control, as described in Procedure 1, acts on the rotation angle of the head. A certain head orientation is performed in the first step of robot turning (Fig. 1(b)). Set point for this control is to maintain this orientation during all the process, by correcting, if necessary, turns suffered by the head when rotating the body. Hence, error signal in this feedback loop is the instantaneous rotation of the head ϕ, which is inferred through SURF flow computation: it searches for the rotation which explains the distortion suffered by consecutive frames in an image sequence. Since the main control variable is the rotation angle, only the horizontal component of the error is considered.

2.2. Rotation angle

Nor artificial landmarks, neither fixed references are used for robot orientation. Hence, from the camera point of view, maintaining the same head orientation during all the process is similar to hold the same camera view (Fig. 2(b)), avoiding image distortions. It will be shown in this section how differences from consecutive frames (i.e. image distortion), computed through the use of SURF flow, allow to extract the rotation angle. Instantaneous rotation is computed from pixel displacements, knowing intrinsic camera parameters (assumed as motion field [1]).

For pure rotations, motion field displays all the vectors pointing almost in the same direction with the same length (Fig. 3(b)). Each one of them captures the distortion suffered by the image due to the camera rotation, because it is a relative change of orientation between the camera and the scene, that is supposed rigid. Moreover, pixel displacements correspond with motion field during instantaneous rotation, since it only depends on 3-D point projections in the image plane (pixel positions) and camera properties. Therefore, instantaneous rotation angle $e^o_{x_k}$ can be computed as the mean of SURF flow vector modules $\bar{M}_k$ and their angles $\bar{A}_k$ (Procedure 1). Afterwards, $e^o_{x_k}$ is cumulated until it can be sent to the controller, and it is used as error signal in the head control.

It could be argued that motion field during a pure rotation shows a very similar configuration to that obtained during a pure translation with only lateral displacement. In both cases, motion vectors are parallel. However, in the later case, their lengths are not the

[1] Control for head rotation is provided by the robot framework `http://www.tekkotsu.org/`

Procedure 1 Head control at instant k

Input: Current image I_k from the camera (Fig. 2(b)), keypoints from previous image P_{k-1}, angular precision pr, horizontal camera resolution res_x, and horizontal opening angle oa_x

Output: Rotation angle: $e^o_{x_k}$

1: **loop**
2: Compute SURF descriptors and keypoint locations of I_k: P_k
3: Find temporal correspondences between P_k and P_{k-1}: M'_k
4: Calculate coarse angles of motion vectors M'_k with precision pr: C_k
5: Use statistical Mode as the most common angle $Md(C_k)$ to refine correspondences: M_k
6: Calculate angles of motion vectors M_k: A_k
7: Compute means of motion vectors M_k and their angles A_k: $\bar{M}_k$, $\bar{A}_k$
8: Define horizontal error in pixels: $e^p_{x_k} = |\bar{M}_k| cos(\bar{A}_k)$
9: Transform error $e^p_{x_k}$ to angles: $e^o_{x_k} = e^p_{x_k} \left(\frac{oa_x}{res_x} \right)$
10: **end loop**

same, but inversely proportional to the depths of the corresponding 3-D points [1]. Since no landmark is considered, keypoint depths are not available and pixel displacements could not be considered a reliable approximation of motion field.

Motion field is not a directly accessible measure, but it is closely related with optical flow under certain circumstances [2]: (1) robot moves on a flat ground, with (2) on-board camera translating in parallel to the ground, and (3) its angular velocity is perpendicular to the ground plane. Unfortunately, for general robots like the one used in this work, constraints do not meet. The Sony Aibo robot is a quadruped robot with a camera on its "nose". Thus, image data is more instable than those provided by a wheeled vehicle with a camera mounted rigidly on its structure. Image instability is due to neck joints, causing head vibrations transmitted to the camera, and specially, for robot walking. Legged robot steps produce very different movements compared to wheeled robot displacements, usually smoother than quadruped robot's gait. Walk behavior in our experiments generates vertical and left-right pendular movements, i.e. camera suffers simultaneous roll and pitch rotations. Only the first assumption out of three is fulfilled in our case. However, since a pure rotation is considered, unfulfilled assumptions will not invalidate the optical flow approximation to motion field.

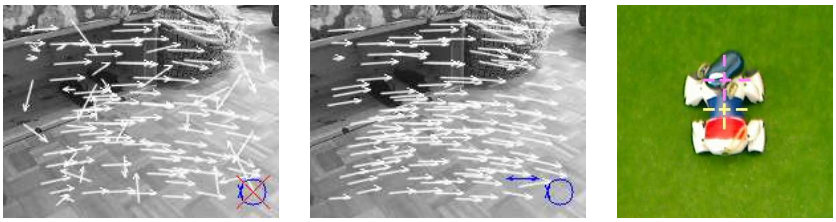
Due to robot configuration, rotation axis of the Sony Aibo robot does not match the axis of the camera rotation, as showed in Fig. 3 (c). This fact introduces an unwanted translation to the initial pure rotation, which will be considered as a perturbation, similar to camera vibration, and it will be assumed to be solved by the controllers. Algorithm introduced for head control ensures the camera orientation will be constant during the process, though the difference between rotation centers will incorporate a translation to the final robot position.

2.3. SURF Flow

SURF flow is defined as 2-D displacements of SURF patterns in the image, where SURF is referred to Speeded Up Robust Features [7]. It is the field resulting from correspondences between SURF keypoints from consecutive frames in a video sequence. Unlike optical flow or the more similar SIFT flow [8], SURF flow is not a dense flow. It is only performed between high confidence keypoints in the image, selected by using a multiscale Hessian detector to find image corners. SURF flow computation is faster than SIFT

(a) (b) (c)

Figure 2. (a) Keypoint correspondences between consecutive images; (b) *Motion vectors* M'_k in the newest image; (c) *Refined motion vectors* M_k (white) with the correspondent *mean vector* $\bar{M}_k$, $\bar{A}_k$ (blue).



(a) (b) (c)

Figure 3. (a) *Motion vectors* M'_k (white) of SURF flow without refinement and *warning* (red) indicating the low confidence of the correspondences. (b) *Refined motion vectors* M_k (white) with the correspondent *mean vector* $\bar{M}_k$ and angle $\bar{A}_k$ (blue). (c) Rotation axis for *head* (pink) and *body* (yellow)

flow, since correspondences are only searched for a few hundreds of keypoints in each image (depending on the image texture), and corner detection and SURF description are computed using Haar wavelets on the integral image representation. Result of this correspondence is shown in Fig. 2(a) and Fig. 2(b).

Moreover, an image correspondence post-processing is applied in order to achieve a better mean vector $\bar{M}_k$ (see Section 2.2). This refinement, showed in Fig. 3, takes place once SURF flow is extracted and the most common angle $Md(C_k)$ is computed, given a certain angle precision pr (see Procedure 1). It consists on search for better correspondences for each keypoint in current image, looking for similar SURF descriptors in a restricted area of previous image. This search area is defined by the triangle ABC, where vertex A is the keypoint in current image, angle $\widehat{BAC} = pr$ defines the search range and the middle point of the edge BC, the triangle size, depends of the velocity of the robot turning. Once correspondences are refined M_k, a more reliable mean vector $\bar{M}_k$ is computed.

Method effectiveness depends, as usual, on assuming that keypoints are found in images, i.e. a textured environment exists. In fact, typical human-made scenes have enough

corners for achieving SURF flow performance. Moreover, SURF flow is robust to optical flow methods' limitations [9]: brightness constancy, temporal persistence or "small movements", and spatial coherence.

3. Results and Discussion

In this section, we present quantitative results of our turn control framework. First, we describe the hardware and software, then, the environment where the test is performed and, finally, experiments are explained.

· *Hardware and Software*: We use the Sony Aibo ERS-7 robot wirelessly communicated with a standard dual-core PC. Experiments are performed using the robot for environment interaction and the computer for hard computation processing. Body alignment is divided in body and head controls. Body control is performed on-board as a reactive behavior, because it acts on rotation velocity of the robot depending on the angle sensor placed in the neck. By contrast, head control is executed in the external computer. Sony Aibo camera captures a 208×159 pixel resolution image and it is sent to the PC every 100ms through wireless connection. The application running on the computer extracts SURF flow from consecutive frames, computing the mean vector and the rotation angle. Then, the angle to turn the head is sent to the robot. Gait behavior for the robot is based on the Tekkotsu software.

· *Environment*: The experiments are performed in an artificial grass surface of about $4m^2$ containing two crossing corridors. It is a natural scenario without artificial landmarks and small variability of the light level. In order to allow a future development in unstructured environments, corridor walls are wallpapered with pictures of real halls and corridors.

· *Experiments*: In order to achieve quantitative results of the system performance, two experiments are defined. In the first one, rotation angle of the head is measured through SURF flow computation and it is compared with pan angle, provided by neck encoder, with the purpose of know the reliability of the SURF flow approximation to motion field. The second experiment consists on measuring the performance of the rotation control proposed in this work, comparing our general approach with the one provided by Aibo Tekkotsu framework.

3.1. Rotation angle

In order to test the reliability of the rotation angle computation, i.e. the confidence of the SURF flow approximation to motion field, the robot head is turned in 5 representative angles. Then, rotation is measured through SURF flow computation and compared with the measure provided by neck encoder: pan angle. Angles are chosen below the middle of horizontal opening angle of the camera ($oa_x/2$) to ensure correspondences among frames, and 30 trials are launched for each one: 3^o, 5^o, 6^o, 10^o, 15^o. Head turning is achieved using the provided module of the robot framework Tekkotsu. The results of this experiment are shown in Table 1. One can see that the values obtained for short angles are promising meanwhile they get worse for angles higher than 6^o. For the three lowest angles both strategies present similar means. However, angle computation through SURF flow has more RMSE (Root Mean Square Error) than a sensor made for the specific task

Table 1. Angle measurements of head rotations by the use of neck encoder and computing the rotation angle through the use of SURF flow

	pan angle		SURF flow angle	
Set point	**mean**	**rsme**	**mean**	**rsme**
3.0000	2.7328	0.3428	3.4359	1.0409
5.0000	4.8138	0.3319	5.5745	2.5608
6.0000	5.0072	1.7533	6.1324	3.1447
10.0000	9.6855	0.4001	4.9603	6.5268
15.0000	14.3472	0.6785	5.6386	9.9232

Table 2. Angle measurements by a zenithal camera of robot rotations. Comparison between *open loop* rotation of a software made for Sony Aibo and *closed loop* rotation for any robot configuration

	Robot framework		General approach	
Set point	**mean**	**rsme**	**mean**	**rsme**
15.0000	11.6128	3.9981	4.7341	10.9956
30.0000	27.6286	4.7443	31.9155	6.7196
45.0000	42.8979	5.8573	41.1000	8.6912
60.0000	63.8203	5.8724	61.1852	9.7878

of sensing angles. This high variability occurs since we assume pure rotation, i.e. if camera axis match the image plane, however the head rotation of the Sony Aibo robot involves a translation. As shown in Fig. 3 (c), the rotation axis is on the neck and the camera is placed on the "nose" of the robot. In this sense, the obtained results for short angles confirm that SURF flow is a reliable approximation to motion field if the error introduced by camera translation does not produce significant changes on the measurements.

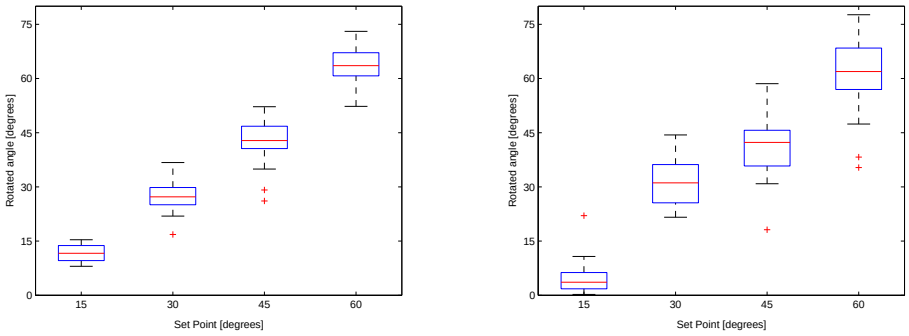
Based on the obtained results, in the turn control experiment we fix the maximum rotation velocity of the robot to 3^o per frame (each 100ms, $30^o/seg$) in other to ensure a reliable sensing of the rotations.

3.2. Robot turning

In order to quantify the performance of the biological inspired rotation control proposed in this work, the robot is turned in four representative angles, and the rotation fulfilled is measured using a zenithal camera. In addition, our general approach is compared with the turning control specifically configured for Sony Aibo, provided by the robot framework Tekkotsu. Angles are chosen below 90^o and 30 trials are launched for each one: 15^o, 30^o, 45^o, and 60^o. 90^o threshold is chosen since it is the maximum turn which the Sony Aibo head can fulfill. In order to measure the rotation angle, colored landmarks are placed in the robot head and tail, and the angle is automatically computed filtering by color the images taken from zenithal camera.

Tekkostu framework provides rotation modules for different robot platforms. However, the open loop control used for this experiment is specifically configured for the Sony Aibo robot. Thus, we used this control to test the performance of the proposed closed loop method, though it can be applied to any robot with a rotation camera. Results of this experiment are presented in Table 2 and Fig. 4.

Results show similar performance for specific Sony Aibo turn control and for our vision based approach, except for 15^o rotation. During short angle rotations, angle be-



(a) (b)

Figure 4. Angle measurements by a zenithal camera of (a) open loop rotation of a software made for Sony Aibo and (b) closed loop rotation for any robot configuration

tween body and head is small, and body control order a low velocity to turn the body. This causes that the robot feet slip and Sony Aibo will remain in the same orientation. For angles higher than 15^o, the proposed rotation approach shows similar or higher mean performance than the specific control provided by robot framework. However, the the vision-based approach shows higher RSME than the robot framework. It is caused by SURF flow approximation to motion field, because of the error introduced by the neck encoder, and possible wireless connection problems.

In particular, some wireless connection problems were observed, losing some frames. When it occurs in consecutive images, the measured angle through SURF flow is not completely reliable and the final angle of the rotation is affected for this loss of information.

4. Conclusions and Future Work

We proposed a biological inspired turn control strategy for robot navigation. The novel approach is exportable to other robotic platforms and configurations, with the only requirement of having a rotating camera. Results shown that turn control is successfully performed without the use of artificial landmarks, taking into account that the robot rotation is a pure rotation, without translational component involved in the movement. The general turning control presented in this work is compared with a specific turn control

for Sony Aibo. In this scenario, our method showed an accuracy as good as the specific control for rotations over 15^o.

Future work will focus on the exportability on different robot platforms and its extension to perform a full vision-based biological inspired framework for path finding, which can involve straight forward and rotation commands. Other improvements include decreasing sampling rate and the duration of actions.

Acknowledgements

This work has been partly supported by projects TIN2009-14404-C02, CONSOLIDER-INGENIO CSD 2007-00018, TSI-020301-2009-27, Spanish Government, and FEDER funds.

References

[1] Trucco, E. Verri, A.: Introductory Techniques for 3-D Computer Vision. Prentice Hall PTR, Upper Saddle River, NJ, USA (1998)
[2] Giachetti, A., Campani, M., Torre, V., CRS, C.: The Use of Optical Flow for Road Navigation. In: IEEE Trans. on Robotics and Automation, vol. 14, pp. 34–48 (1998)
[3] Nagatani, K., Tachibana, S., Sofne, M., Tanaka, Y.: Improvement of odometry for omnidirectional vehicle using optical flow information. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, pp. 468–473 (2000)
[4] Sorensen, D.K., Smukala, V., Ovinis, M., Lee, S.: On-line optical flow feedback for mobile robot localization/navigation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 1246–1251 (2003)
[5] Yoon, K., Jang, G., Kim, S., Kweon, I.: Color landmark based self-localization for indoor mobile robots. J. Control Autom. Syst. Eng. 7, 9, 749–757 (2001)
[6] Deng, X. and Milios, E. and Mirzaian, A.: Landmark selection strategies for path execution. Robotics and Autonomous Systems 17, 171–185 (1996)
[7] Herbert, B., Tinne, T., Luc, VG.: Surf: Speeded up robust features. Lecture Notes in Computer Science 3951, 404–417, Computer Vision–ECCV 2006 (2006)
[8] Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.: SIFT flow: dense correspondence across different scenes. Lecture Notes in Computer Science 5304, 28–42, Computer Vision–ECCV 2008 (2008)
[9] Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI, vol. 3, pp. 674–679, Citeseer (1981)
[10] Grasso, R., Assaiante, C., Prevost, P., Berthoz. A.: Development of anticipatory orienting strategies during locomotor tasks in children. In: Neuroscience & Biobehavioral Reviews, vol. 22, pp. 533–539, Elsevier (1998)
[11] Grasso, R, Prevost, P, Ivanenko, YP, Berthoz, A.: Eye-head coordination for the steering of locomotion in humans: an anticipatory synergy. In: Neuroscience Letters, vol. 253, pp. 115–118, Elsevier (1998)
[12] Hicheur, H., Vieilledent, S., Berthoz, A.: Head motion in humans alternating between straight and curved walking path: combination of stabilizing and anticipatory orienting mechanisms. In: Neuroscience letters, vol. 383, pp.87–92, Elsevier (2005)
[13] Tran, M., Soueres, P., Taix, M., Girard, B.:. Eye-centered vs body-centered reaching control: A robotics insight into the neuroscience debate. In: Robotics and Biomimetics (2009)
[14] Perez-Sala, X., Angulo, C., Escalera, S.: Biologically Inspired Path Execution Using SURF Flow in Robot Navigation. In: IWANN (2011)

Determining Where to Grasp Cloth Using Depth Information

Arnau RAMISA [a,1], Guillem ALENYA [a], Francesc MORENO-NOGUER [a] and Carme TORRAS [a]

[a] *Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona*

Abstract. In this paper we address the problem of finding an initial good grasping point for the robotic manipulation of textile objects lying on a flat surface. Given as input a point cloud of the cloth acquired with a 3D camera, we propose choosing as grasping points those that maximize a new measure of wrinkledness, computed from the distribution of normal directions over local neighborhoods. Real grasping experiments using a robotic arm are performed, showing that the proposed measure leads to promising results.

Keywords. Deformable Objects, Computer Vision, Grasping point selection

Introduction

Recently the problem of grasping and folding clothes with a robotic arm has attracted much attention [1,2,3,4,5,6]. Its application ranges from automating industrial cleaning facilities to domestic service robots. There exist works devoted to determining the best/optimal grasping point for a particular purpose (e.g. folding) once the cloth is held by a robotic hand. However, most of the research in this area has been carried out in controlled environments and simple heuristics have sufficed.

In [3] the authors designed a cloth grasping-point selection system to autonomously take elements form a pile of washed towels and fold and stack them ready for storage using a Willow Garage PR2 robot. The main contribution of the paper in terms of grasping is being able to detect a corner when the towel is already grasped by one of the robot arms, and the initial pick-up is done by selecting the central point of the cloth, detected through background segmentation and stereo. The performance of this initial grasp was not reported in the paper.

In [4] the authors describe a complete system, designed for the PR2 robot, for laundry folding. The system can start from a pile of clothes, pick up and identify one of them and bring it into a desired configuration; and repeat the procedure until no more clothes are left. Two HMM are used for the two tasks, identification and manipulation. The cloth is initially grasped by one edge. This kind of grasping with PR2 manipulators is only possible because the surface in which the cloth lies is made of a soft material that deforms under the robotic hand.

[1]Corresponding Author: Arnau Ramisa, Institut de Robòtica i Informàtica Industrial, CSIC-UPC. Parc Tecnològic de Barcelona. C/ Llorens i Artigas 4-6. 08028 Barcelona. E-mail: aramisa@iri.upc.edu.

A complete system for retrieving one by one all elements of a laundry basket or pile, classifying and then folding them is proposed in [6]. In this approach, the topmost element of the pile is found using stereo vision, and its geometric center is used as grasping point. The grasping operation is repeated as many times as necessary to assure a correct grasp.

Foresti and Pellegrino presented a vision-based system to detect grasping points for furs in an industrial environment [7]. Their system used a hierarchy of Self-Organizing Maps to segment the image into fur and non-fur areas, and then analyzed the detected fur blobs to determine the best picking point for the robotic arm.

Taking a more sophisticated approach, in [5] the authors propose a method for estimating the pose of cloth through parametrized shape models, specific for each category of clothing (e.g. t-shirt, pants, towel). These models were fit using an energy optimization approach, and then used to classify the cloth item type prior to folding it with an open loop movement sequence. Finding an accurate cloth item pose model would definitely help in the task of initial grasping point selection; however, fitting the model to a cloth in each image acquired by the robot can take up to 2.5 minutes when running in a multi-core laptop computer, which may be too expensive for practical applications. Furthermore, to fit the required models the cloth has to be presented to the system in a canonical form, only possible thanks to a previous manipulation step.

In this paper, our purpose is to investigate what constitutes a good initial grasping point for a piece of cloth lying on a flat surface in an arbitrary configuration. We propose a new "wrinkledness" measure based on range information that can be used to determine the most easily graspable point at an affordable computational cost.

Recently, a method related to ours that detects cloth objects using 2D images has been proposed [2]. A cloth detection method in a domestic or office environment based on wrinkle features is presented. The wrinkle features were found by analyzing the response of Gabor filters with a Support Vector Machine (SVM) trained with manually annotated images of wrinkled cloth. Finally, graph cuts were used to segment the cloth pieces from the background. The proposed method was applied in a cloth pick-up task using a mobile robot, which selected its grasping target as the wrinkle with maximal 3D volume according to stereo camera measurements.

Although this approach is very appealing, it differs from ours in a number of points: First, although our approach could be used for the same purposes, [2] tackles the more comprehensive task of detecting clothes in an unprepared environment, and not the identification of optimal grasping points. Second, we directly use 3D information obtained from an economic sensor (therefore avoiding the expensive data collection and manual annotation step for training the SVM), and thus our procedure is not vulnerable to any learning error.

In order to acquire the depth information, we rely on a Kinect 3D camera[2], which is a very affordable device to obtain simultaneously depth and texture maps in an indoor environment, and is becoming of widespread use in the robotics and computer vision communities, directly competing with the much more expensive time-of-flight PMD cameras [8]. Kinect uses an infrared structured light emitter to project a pattern into the scene and a camera to acquire the image of the pattern, then depth is computed by means of structured light algorithms. Additionally, among others sensors, the Kinect integrates a high resolution color camera.

[2]Developed by Prime Sense `http://www.primesense.com/`

Figure 1. Textured point clouds (left) and images (right) acquired with the Kinect 3D camera. As can be seen, some points are missing due to lack of resolution and occlusions in the wrinkled point cloud.

Kinect was developed with the idea of robust interactive human body tracking and great efforts have been made in that direction [9]. After the Kinect protocol was hacked, the community rapidly started to use it, first with the same idea of human interaction and afterward in other areas, like robot navigation. Later, the official library was made public through the OpenNi organization.

Some pictures and point clouds of a towel acquired with the Kinect 3D camera are shown in Figure 1. As can be seen, the resolution is not sufficient to perceive small details, and some holes without depth information occur.

1. Graspable Point Detection

In this section we first present in detail the problem that we are addressing and the dataset collected to validate the method. Next, we describe the proposed algorithm to determine a graspable point given an input image and a depth map of a cloth.

Our initial assumption is that a good grasping point for a textile object lying on a table is one where the cloth defines ridges or other 3D structures, i.e. where there are wrinkles. The justification of this assumption comes from the nature of the grasping mechanism, which in our case has three fingers, with a total of four degrees of freedom. Lacking the precision of movement, flexibility and the small(er) size of human hands

(which can pick up cloth objects from the edges), the best point for a grasp with this type of hand is a pyramidal or conic-like shape, such as the one produced by wrinkles.

One common heuristic or workaround used by works addressing textile grasping such as [3,6] is to select as grasping point the highest one in the 3D point cloud of the cloth object. In practice, the highest point usually coincides with a wrinkle, and thus it is in agreement with the assumption stated above. However, the highest point is not necessarily a good grasping point in all situations, as we show in the experimental section of this paper (see Section 2). The objective of this work is to explore how to characterize the "wrinkledness" of a cloth object, and to find out its advantages and drawbacks for detecting good grasping points. Ultimately, this cue can be used in the design of a method able to determine the best grasping point in a robust way.

1.1. Proposed Wrinkledness Measure

We have developed a measure of the "wrinkledness" in a point taking into account the depth information of its neighborhood. This measure is computed using a local descriptor based on the surface normals of a 3D point cloud. In particular, we use the *inclination* and *azimuth* angles in the spherical coordinates representation of the normal vectors:

$$(\phi, \theta) = \left(arccos\left(\frac{z}{r}\right), arctan\left(\frac{y}{x}\right)\right) \tag{1}$$

where ϕ is the inclination and θ is the azimuth, (x, y, z) are the 3D point coordinates, and r is the radius in spherical, defined as:

$$r = \sqrt{x^2 + y^2 + z^2}. \tag{2}$$

Next, we model the distribution of the inclination and azimuth values in a local region around each point. Although it would be very interesting to introduce spatial subdivisions on the local region and model the distribution of the angles locally in each subdivision, for our current purposes we found it was not necessary and therefore we left it as future work.

We evaluated two possibilities for modeling the spherical coordinate angles distribution: two sixty-four-bin histograms, one for each angle, and a single two-dimensional histogram with 64×64 bins that considers both angles jointly.

A beneficial side effect of this process is that occluded regions and areas where the Kinect was not able to estimate the depth are naturally interpolated using the information provided by their neighbors, which reduces the sparsity of the point cloud.

From this model of the local distribution of normal angles in spherical coordinates, we seek to estimate the "wrinkledness" of a point. This can be intuitively done by looking at the spread of the angle histogram: the more different orientations the surface takes, the more likely that it is a highly wrinkled area. Although standard deviation is probably the first measure of spread that comes to mind, it is not a good choice, since a strongly bimodal distribution can have a large standard deviation while having low spread. A better choice is entropy, which does not suffer from this drawback:

$$H(X) = -\sum_{i=1}^{n} p(x_i)\ log\ p(x_i) \tag{3}$$

where X is the n-bin angle orientation histogram, and x_i is the ith bin.

Entropy measures how much information exists in a message or distribution, or alternatively, how "predictable" it is. In our context, it directly tells us the amount of support of the distribution concentrated in high probability peaks or, equivalently, how much of the surrounding area of the point has normals aligned in the same orientation i.e. a flat surface or a combination of a few flat surfaces. Entropy has the additional advantage of not assuming an unimodal distribution like standard deviation or high-order moments such as kurtosis and skewness.

In Figure 2 "wrinkledness" maps using the proposed entropy-based measure can be seen for the 1D and the 2D histogram representations. It can be observed that the 2D measure produces slightly clearer maps with a longer range of activation levels, faithfully reflecting the wrinkled areas of the towel. Of course it is possible to vary the support region of the descriptors (or downsample the point cloud). This would allow to obtain a smoother result which only reflects a few "global" maxima with large support areas, and peaky "wrinkledness" maps capturing all the local maxima when a small support region is used. Figure 3 shows the response obtained with different support regions. One important limitation of this approach is that concave areas of the image get a high activation level while not being a good grasping point. Yet, it is possible to compute a concavity measure and use it to re-weight the "wrinkledness" map. Finally, the peaks of the map can be used as candidate grasping points.

2. Experiments

We tested our proposed "wrinkledness" measure in real grasping experiments. Our experimental setup consists of a robotic hand with three fingers installed in front of a flat table of uniform color, in which the cloth object was positioned. We mounted a Kinect 3D camera having a zenital view of the table and providing the information used to select the grasping point. The piece of cloth used in the experiments is the previously seen small red towel which, for the purpose of evaluating our method, is segmented from the background using a simple color thresholding procedure, as done in similar works.

Five experiments were performed with different initial configurations of the towel. In all cases, the 2D histogram with a square support region with a side of 33 pixels was used to generate the "wrinkledness" map after segmenting the towel from the table, and the point with the highest activation was selected as the grasping point. Next the robotic arm was moved to the point, and a grasp attempt was performed. Please note that we are not claiming that the point with highest activation in the map is necessarily the best grasping point. However here we used this simple heuristic with very good results. Four out of five tests ended with a successful grasp. Figure 4 shows the images and "wrinkledness" maps used to decide the grasping point, and a photo of the robotic arm holding the towel for those tests that were successful. In each successive test the towel was positioned in an increasingly difficult configuration.

- In the first test, there was one clear peak in the center of the elsewhere flat towel, which was correctly detected and ended with a successful grasp.
- For the second test, the towel contained a similar wrinkle, but this time at the corner, making the task more difficult (smaller support region, less clear wrinkle).

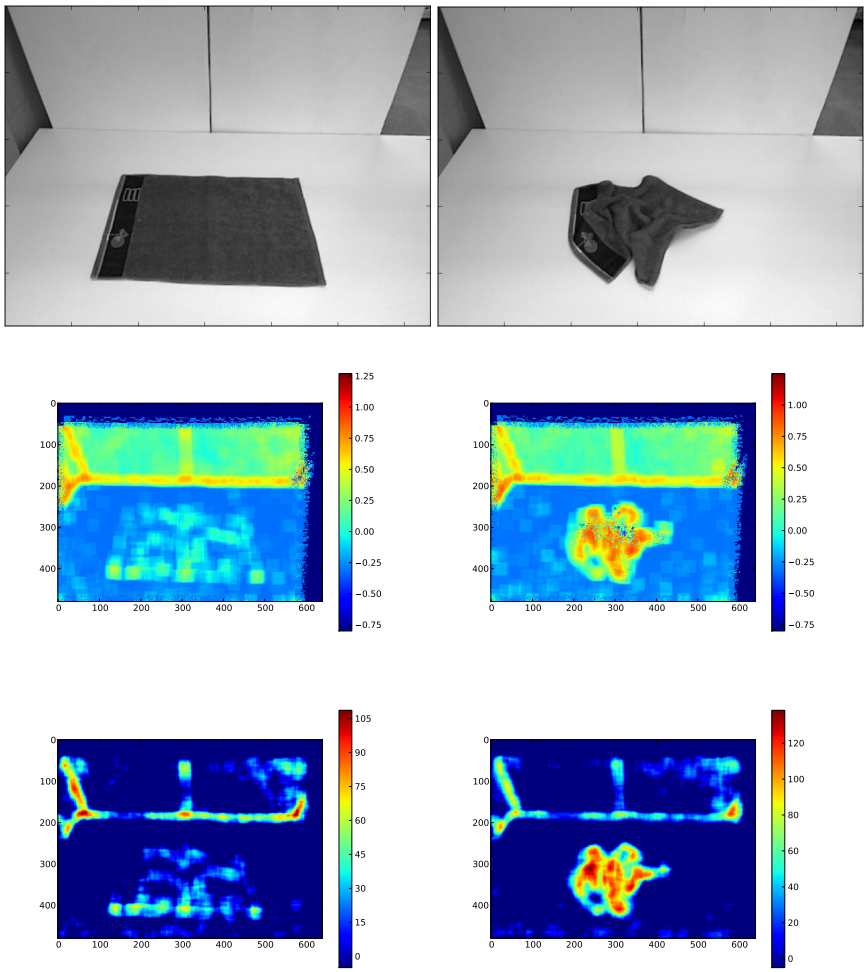


Figure 2. Response of the proposed "wrinkledness" measure. First row: original images, second row: "wrinkledness" map for the 1D histograms measure, third row: "wrinkledness" map for the 2D histograms measure. The considered local area around each point is of 33 pixels. Please note the difference in the colorbar range.

However, the method was able to find a good grasping point and the experiment was successful.

- The third test exemplifies our motivation to characterize good grasping points. In this test, the towel presents a configuration in which the highest point is not good for grasping (it is an almost flat surface), but a lower point is. The presented wrinkledness measure finds the right (lower) point and the grasp ends successfully.
- In the fourth test, most of the area of the towel is concealed by a fold, and only two small wrinkles are present in the uncovered area. Our method does not find the best grasping point from a human perspective, but the selected point is not

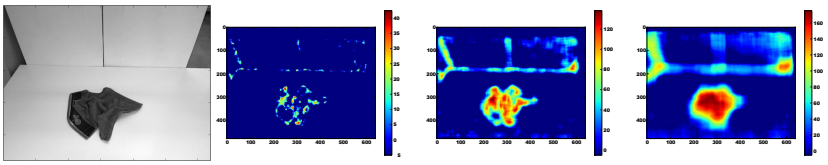


Figure 3. Response obtained with different support regions for the local descriptor. From left to right: Original image, "wrinkledness" maps with support region 15×15, 33×33 and 65×65.

completely bad either and the grasp ends successfully as well. The confusion of the method in this case is due to the merging of two different layers of cloth in the same local region. A drawback of selecting this kind of points is that it can lead to grasping the towel from two separate points at a time, making it more difficult to succeed in the subsequent tasks (e.g. grasping the towel from the edges for folding) to succeed. Introducing a continuity-enforcing measure to the local region would help to prevent this situation.

- Finally, in the last experiment, the robotic hand did not find any graspable surface in the selected point. The failure is due to a concavity being detected as the point with highest entropy in the orientation of the normals. As mentioned earlier, for the proposed measure to be more robust, concavities should be detected and down-weighted in the "wrinkledness" map.

3. Conclusions

In this paper we have presented some preliminary work towards finding a good measure of "graspability" for cloth objects lying on a flat surface. This is an important aspect for making robots fully autonomous in unprepared environments; in contrast, related literature so far relied on simple heuristics, that worked in controlled settings.

Our proposed measure is computed from point clouds acquired with a Kinect 3D camera, and uses entropy in the normal vector orientation distribution around a given point as an indicator of wrinkledness. Wrinkled areas constitute, in general, good graspable points.

Although not addressed here, this measure can be later combined with other cues to make it more robust. In particular, concave areas pose a problem since they yield a high score with our measure, but they are in general not good for grasping.

We performed real grasping experiments with the proposed measure. For them, a towel placed on a table and observed by a Kinect 3D camera from a zenital position, was grasped from the point of maximum entropy in the normal directions of the towel surface. The towel was previously segmented from the table using color. Four out of five tests ended in a successful grasp, showing that the presented measure is suitable to be used for identifying good grasping points.

In terms of computational cost, our non-optimized implementation is able to compute a dense "wrinkledness" map in a few seconds. Moreover, it is possible to make the grid more sparse without affecting too much the quality of the results (e.g. computing

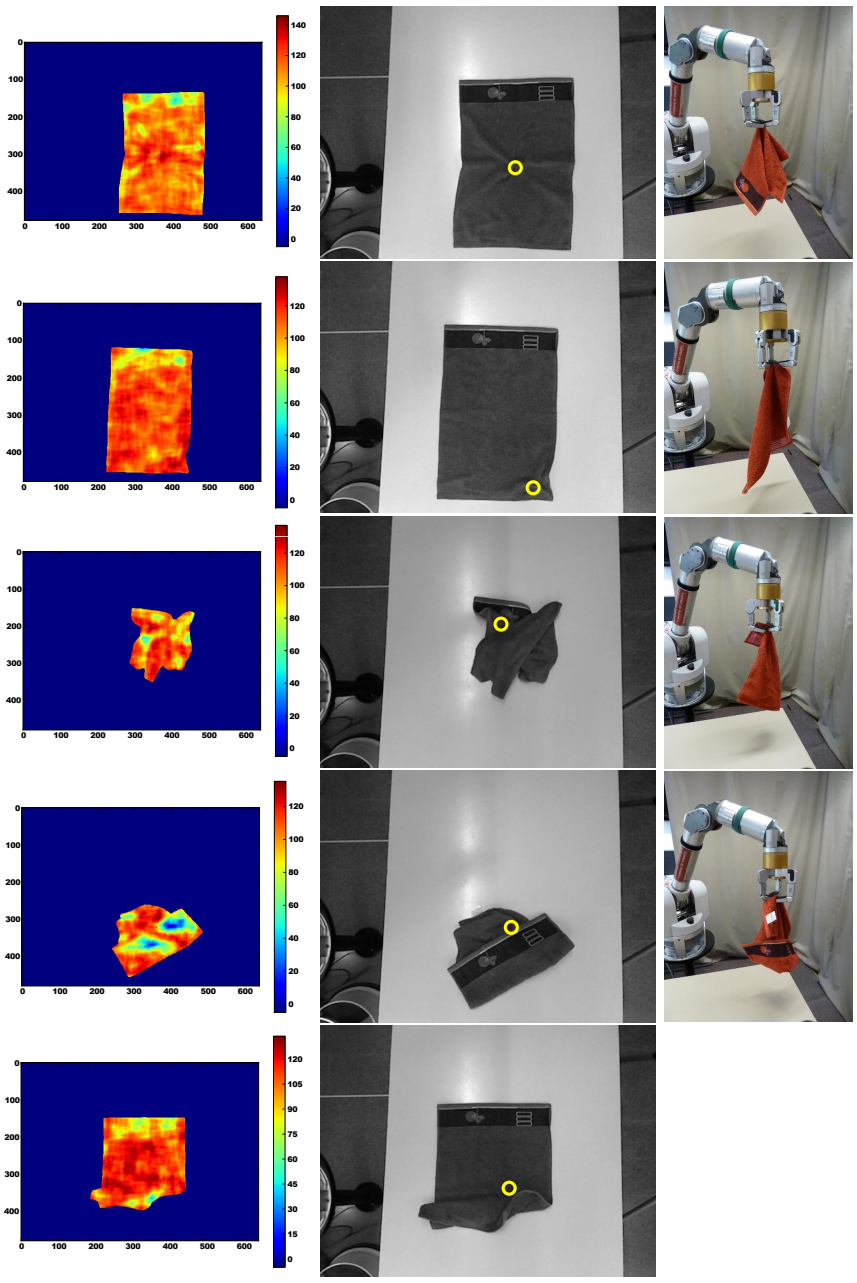


Figure 4. Details of the five experiments conducted with a robotic arm (one per row). For each experiment the following are shown (in order): the segmented "wrinkledness" map of the towel, the selected grasping point, and a picture of the robotic hand with the grasped towel, if successful.

the entropy at every three pixels instead of at every pixel) would significantly reduce the computation time.

Future work includes more thoroughly evaluating the proposed measure to identify its weaknesses, and researching other cues that can help make it more robust.

One point worth exploring is a concavity detector to avoid selecting points where no graspable surface can be found. While simple algorithms can suffice for this task, they will provide a considerable improvement of the final "graspability" measure we are pursuing.

Another interesting future work would be investigating how to improve the normal orientation descriptor by computing it in subdivisions of the local region around the point, using soft voting to reduce the effect of aliasing occurring at orientation bin boundaries or varying the number of orientation bins. Moreover, taking advantage of a multi-scale representation of the "wrinkledness" map to better characterize optimal grasping point locations would be interesting as future work.

Finally, better grasping points could be found by combining information like point height, total 3D volume, normal orientation or the aforementioned concavity measure with the entropy-based measure proposed in this paper.

Acknowledgements

This research is partially funded by the Spanish Ministry of Science and Innovation under projects DPI2008-06022 and MIPRCV Consolider Ingenio CSD2007-00018, and the Catalan Research Commission.

References

[1] F. Osawa, H. Seki, and Y. Kamiya, "Unfolding of massive laundry and classification types by dual manipulator," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 11, no. 5, pp. 457–463, 2007.

[2] K. Yamakazi and M. Inaba, "A cloth detection method based on image wrinkle feature for daily assistive robots," in *IAPR Conf. on Machine Vision Applications*, pp. 366–369, 2009.

[3] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 2308–2315, IEEE, 2010.

[4] M. Cusumano-towner, A. Singh, S. Miller, J. F. O. Brien, and P. Abbeel, "Bringing Clothing into Desired Configurations with Limited Perception," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, (Shangai, China), pp. 3893–3900, 2011.

[5] S. Miller, M. Fritz, T. Darrell, and P. Abbeel, "Parametrized Shape Models for Clothing," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, (Shangai, China), pp. 4861–4868, 2011.

[6] B. Willimon, S. Birchfield, and I. Walker, "Classification of Clothing using Interactive Perception," in *Proc. IEEE International Conference on Robotics and Automation (ICRA11)*, pp. 1862–868, 2011.

[7] G. Foresti and F. Pellegrino, "Automatic Visual Recognition of Deformable Objects for Grasping and Manipulation," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 34, pp. 325–333, Aug. 2004.

[8] S. Foix, G. Alenyà, and C. Torras, "Lock-in Time-of-Flight (ToF) cameras: a survey," *IEEE_J_SENSOR*, vol. to appear, 2011.

[9] J. Shotton, A. Fitzgibbon, M. Cook, and T. Sharp, "Real-time human pose recognition in parts from single depth images," in *Computer Vision and Pattern Recognition (CVPR)*, p. to appear, 2011.

Self-Supervised Clustering for Codebook Construction: An Application to Object Localization

Arturo RIBES [a,1] Senshan JI [b] Arnau RAMISA [a] Ramon LOPEZ DE MANTARAS [a]
[a] *IIIA-CSIC, Campus Universitat Autònoma de Barcelona*
[b] *Departament de Matemàtiques, Universitat Autònoma de Barcelona*

Abstract. Approaches to object localization based on codebooks do not exploit the dependencies between appearance and geometric information present in training data. This work addresses the problem of computing a codebook tailored to the task of localization by applying regularization based on geometric information. We present a novel method, the Regularized Combined Partitional-Agglomerative clustering, which extends the standard CPA method by adding extra knowledge to the clustering process to preserve as much geometric information as needed. Due to the time complexity of the methodology, we also present an implementation on the GPU using nVIDIA CUDA technology, speeding up the process with a factor over 100x.

Keywords. Self-supervised clustering, agglomerative clustering, object localization

Introduction

Visual word representations have become a very popular and successful approach, adopted by many state of the art object recognition methods. This is mainly due to their capacity to enable feature sharing [15] and aggregating statistics in a local region of the feature space to build robust probabilistic models. Object recognition methods based on visual words rely on the construction of a codebook of appearance clusters which quantize some high-dimensional feature space. This codebook is later used to map any visual feature to a finite set of primitives, suitable for machine learning techniques that reduce the impact of the well-known curse of dimensionality.

Recently several works have addressed the introduction of extra knowledge into the codebook representations at different levels. In [2] it is done during the clustering process itself. The Information Bottleneck Method [13] is introduced in a partitional clustering scheme. First, an overdiscretized partition of the feature space is constructed and then, a radius-based clustering method is used to obtain a very discriminative representation with only a few visual words, adapted for bag-of-words representation. Nevertheless, the

[1]Corresponding Author: Arturo Ribes, IIIA-CSIC, Campus Universitat Autònoma de Barcelona, E-08193, Bellaterra, Catalonia, Spain; E-mail: aribes@iiia.csic.es.
This work was supported by the JAE-PREDOC grant from the CSIC and by the grant 2009-SGR-1434 of the Generalitat de Catalunya

contribution of [2] is a reduction of computational cost for a pixel-level labeling method, and not for improving its accuracy. Other approaches construct intermediate features by searching for visual and spatial configurations that occur frequently [10] or that are very discriminative for a given object class, either by its visual or spatial properties [18]. Semantic vocabularies are also constructed using manually annotated ground truth data, where meaningful labels are assigned to regions of the image (e.g. sky, building, etc.) [17]. In [9], specific vocabularies for particular concepts are derived from a universal one.

In [4], the codebook construction is split into three stages: Clustering by visual similarity, co-location, and finally co-activation. A radius-based clustering scheme – the Combined Partitional-Agglomerative (CPA) clustering – is used, changing the measure that estimates cluster similarity. In the first stage, a set of cluster representatives used for matching appearance features are constructed. The other two stages work at a semantic level, modeled by a Bayesian *part-of* network. In the co-location clustering stage, clusters that occur in different images in roughly the same locations are merged, building semantic sub-part clusters. Finally, the co-activation clustering merges sub-part clusters that occur in nearby locations in the same image, building part clusters, which provide the evidence for object presence. Very good results are achieved with this method, cutting down the number of nodes that need to be used for belief computation. For a more extensive comparison between compact codebook construction methods, the interested reader is referred to [16].

In this work, we contribute a novel method, the Regularized CPA clustering, which adds extra knowledge to the clustering process to preserve as much geometric information as needed. This method improves the standard CPA clustering method by introducing a regularization term based on spatial information, which provides the property of self-supervision to the clustering method. The resulting clusters have more discriminative power in estimating object locations in novel images than using the standard method.

The rest of this paper is structured as follows: in Section 1 the base object localization framework used in this work is explained. Section 2 reviews the main codebook construction methods used in the related literature. Next, in Section 3 the proposed method is described and in Section 4 the experimental setup and the obtained results are presented. Finally, in Section 5 we draw conclusions and propose the future work.

1. Generalized Hough Transform Object Localization Approach

Approaches to object localization based on the Generalized Hough Transform (GHT) [5,6,7] work by accumulating votes coming from each local descriptor or visual word present in the image in a pose parameter space, in which later one can search for local maxima to obtain feasible object hypotheses. The main steps of the process are depicted in Figure 1. The initial stages are similar to those of the standard histogram-based bag of features approach [1], as we also use local features and a codebook of visual words. After the appearance codebook is built, we compute the geometric distributions of visual words on objects. Here we use a star-shaped model, as in [5,7], which assumes that feature occurrences are independent given the object centroid. The geometric space is parametrized in three dimensions, two for locations and the scale, representing the offset from feature location to object centroid, thus, a feature-centric view. In our experiments

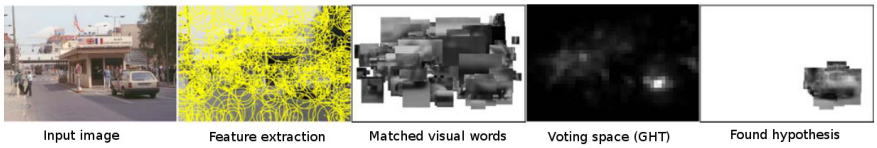


Figure 1. Steps of the standard object localization approach using the GHT.

we used scale- and rotation-invariant features, so offsets are rotated back to the canonical orientation to obtain 3-D normalization. In [5], geometric distributions are modeled as point samples with a constant weight, while in [7] the geometric space is discretized with a grid of the same size as the GHT used in the detection stage, allowing to compute a weight on each cell of the grid as the log-likelihood score of object presence given feature presence. In our experiments we evaluated both approaches.

In order to localize an object in a novel image, local features are extracted and matched to the appearance codebook to obtain the visual words. Then, each of these visual words casts votes in the GHT based on its geometric distribution. After all votes have been cast, the local maxima in the voting space forms the initial set of hypotheses. As typically a high number of false positives is found in this set, a refinement stage is necessary. In [5] good results are obtained with a scale-adapted version of the Mean-Shift algorithm to refine the hypotheses, followed by a final Minimum Description Length (MDL) verification stage. In contrast, [7] learns an optimal threshold based on a MAP estimate by modelling positive and negative hypothesis scores as Gaussian distributions. In our experiments, we took this last approach for simplicity, although the reported results for MDL show good improvements. However, this work focuses on the clustering process, which is not related to the choice of the hypotheses validation method. This pipeline has been widely adopted by the research community, with many variations that address particular issues, mainly dealing with multiple views of the objects [12], the huge amount of false positives that arise from this bottom-up approach and enhancing the model with Bayesian inference capabilities [4].

2. Codebook Construction Methods

Our work focuses on optimizing the codebook learning process, tailoring it to the task at hand by introducing information from the geometric distribution of the features. Good clustering methods make the object localization schema more stable and less codebook entries have to be activated during the matching stage, as shown in [3]. Next we will briefly review the two main clustering techniques (partitional and agglomerative) used for codebook learning in recent related literature.

From the two families of clustering schema, the most frequently used methods are k-Means, its hierarchical variant and Gaussian Mixture Models (GMM) as partitional methods and Single- or Average-link Clustering as agglomerative methods, being the last one also known as Unweighted Pair Group Method with Arithmetic Mean (UPGMA). Partitional methods are based on the EM pattern, where each data point is assigned to the closest cluster and the cluster representatives are recomputed, repeating the process until the convergence criteria are satisfied. The drawbacks of partitional methods are the predefined number of clusters, the boundary artifacts, its dependence on a good initial-

ization of cluster representatives (although good initialization methods for k-Means have been proposed) and that the methods are typically biased to put more clusters in very populated regions of the feature space [16].

Agglomerative methods, on the other hand, start with as many clusters as training data points and, at each iteration, the two closest clusters are merged until the convergence criterion is met. The usual stopping criterion is a threshold on the maximum distance between the closest pair of clusters, which determines how compact our clusters will be. Besides, the clustering trace can be saved and revisited to generate clusterings with different thresholds at almost no computational cost. The main bottleneck of agglomerative clustering is the distance matrix computation, which has squared time and space complexity. With some minor modifications, the space complexity can be reduced to linear storing only the closest neighbor for each cluster. In [3] a method to reduce the computational complexity is proposed, which consists in maintaining a table of cluster Reciprocal Nearest Neighbors (RNN) and, after each merge, update only the table for neighbors of the merged clusters. Furthermore, the distances between clusters can be efficiently determined using its mean and variance, which can be incrementally computed. However, this method works only if the clustering criterion fulfills the reducibility property:

$$d(c_i, c_j) \leq inf(d(c_i, c_k), d(c_j, c_k)) \Rightarrow inf(d(c_i, c_k), d(c_j, c_k)) \leq d(c_i \cup c_j, c_k) \quad (1)$$

where c_i,c_j and c_k are clusters and $d(c_j, c_k)$ is a distance measure. This has been proven to be valid for the Average-link criterion using Euclidean distances but, as it will be shown, it is not valid for the regularized inter-cluster distance used in our experiments. With these modifications, agglomerative clustering is suitable for clustering large sets of data. In [3] is shown that the run time of k-Means exceeds that of agglomerative clustering using RNN for more than ten or twenty thousand data points. Furthermore, clusters resulting from agglomerative methods have lower variance, and therefore less ambiguity when matching novel features to codebook entries.

3. Object Localization with a Regularized Codebook

Our proposed object localization system is based on the Generalized Hough Transform based approach for object localization described in section 1, but with some improvements, mainly in codebook creation. The feature detection stage relies on the Harris-Laplace detector, which is known to give state of the art results in object categorization. Extracted patches are described in a rotation- and scale-invariant frame using the well known SIFT descriptor. After all features have been extracted from training images, we construct the codebook using the method proposed in this section. Finally, for matching new SIFT descriptors to codebook entries, hard-assignment is used: We assign the identity of the closest cluster prototype. In [5] authors show that soft-assignment in recognition mode does not provide an accuracy improvement, but it does for learning. We expect that using our regularized clustering scheme, soft-assignment will not be necessary even in the learning stage. Due to space limitations, we describe in detail only the regularized codebook construction method, and readers interested in a more comprehensive explanation of our complete method are referred to [11].

3.1. Combined Partitional-Agglomerative clustering

In [7], the codebook generation is efficiently computed using k-Means clustering of Gabor jets and color histograms. However, as stated before, it has been shown that agglomerative clustering gives better codebooks in terms of cluster compactness. However, agglomerative clustering is very demanding in memory and computation if our data set is bigger than about $20,000$ samples. This is a problem if we want to cluster hundreds of thousands of features, as happens with object category recognition with multiple classes. In order to address this limitation, we have adopted the schema known as Combined Partitional-Agglomerative (CPA) clustering. It consists in applying a partitional method (e.g. k-Means) to the whole data set, obtaining k partitions of no more than $20,000$ samples. Next, agglomerative clustering is applied to each partition independently using a low distance threshold, so we end up having k relatively small sets of clusters. Then, all the k sets are combined and agglomerative clustering is applied again, this time using the distance threshold that we would normally use. The resulting cluster representatives form our codebook.

3.2. Regularized CPA clustering

A careful reader may have noticed that using the clustering schema described above, the stopping criterion used – the threshold on inter-cluster distance – may be too weak for obtaining a good representation suitable for the object localization task. To solve this, we propose to bring extra knowledge into the clustering process to obtain a more discriminative discretization of the feature space. As we saw in the object detection pipeline, appearance patches are extracted from the image, matched to a codebook and then the geometric distributions of activated codebook entries are used to accumulate evidence of object presence in a given location on the image. In the optic of scale-space theory, it is desirable a trade-off between the accuracy of the geometric distributions associated with codebook clusters and the aperture of its receptive field in the appearance space.

The information bottleneck method has been recently used in many successful approaches ranging from feature selection [14] and codebook refinement for bag-of-words representations [2]. This principle maximizes the following Lagrangian:

$$I(C;Y) - \lambda * I(C;X) \tag{2}$$

where X is the random variable we want to code, Y is a relevant variable and C is the compressed variable, the codebook. In our application, we want that the codebook representatives lose as much information as possible from the initial feature set, that is, we want the clustering process to maximize the aperture of the receptive fields in SIFT appearance space, while preserving a fraction – governed by the λ parameter – of the information shared between each codebook entry and its spatial distribution relative to object centroid. In the standard procedure, the best merge is the pair of clusters which have the minimum inter-cluster distance. This is very simple and works well in the initial stages of clustering, where C is still very close to X (Eq. 2) and the optimal merge is obvious. At some point of the iterative merging procedure, selecting the best merge becomes harder, as there are many pairs of clusters with roughly the same inter-cluster distance. If the task is to make good predictions for object locations, it would be a bad choice to merge two similar clusters in appearance space but very different semantically, e.g. a cluster

representing eye shapes and another for mouth shapes. With the aim of evidencing this effect, we computed the distance between the spatial center of the hyper-rectangle encompassing all the clusters and the density center. If the distribution is uniform, then the density center should coincide with the spatial center. The measure is defined as:

$$degree_{uni} = ||x_{spatial} - x_{density}||_2$$

where $x_{spatial} = \frac{x_{min}+x_{max}}{2}$ and $x_{density} = \frac{1}{N}\sum x_i$

This contingency calls for regularization, so in our method we added such a term based on the distance between the underlying geometric distributions linked to the pair of clusters to be merged. In our experiments, we used a metric based on a symmetrized estimate of the Kullback-Leibler divergence between the two geometric distributions, which gave better results compared to Chi-square.

Now we have explained *how* we obtain good merge candidates, it remains to explain *when* we need to compute them. Ideally this should be done from the start, but there are some problems with that: First, KL divergence estimate is based on the k-NN framework, so we need enough samples to get a good probability density estimate; second, at the initial stages of clustering the merges are evident, so only with appearance distance is enough; and last, but most important, the regularized inter-cluster distance does not fulfill the reducibility property that RNN needs, so we need to recompute the distances at each iteration, making the regularized clustering prohibitively slow if applied too early. The solution we propose goes through defining two parameters: The first parameter is the applicability threshold T_{reg}, which dictates the minimum inter-cluster appearance distance necessary to start computing the regularization term, and the second is the minimum number of geometric samples a distribution needs to contain to be considered as object and not background. The optimal choice for the first parameter is determined experimentally, and follows a better explanation of the second. A cluster is labeled as being either object or background depending on the number of geometric samples to compute a spatial distribution that it contains. This is related to cluster precision in object-background terms. Note that by geometric samples we mean features found inside a ground truth bounding box belonging to an object while training, which are the only ones used to estimate object centroids. With this concept in mind, we can distinguish three kinds of merges:

- Object vs. object. This is the only case where we can actually compute the geometric distance between corresponding spatial distributions.
- Background vs. background. In this case, computing the geometric distance is not meaningful, but still we have to define a value for it.
- Object vs. background. This case is the most problematic, as can have two possible interpretations: either it is a real background cluster, so we do not want to merge them, or it is an object cluster that still has not accumulated enough geometric samples. As in the previous case, we need to define a value for the geometric distance.

For second and third kind of merges, the most straightforward solution is assigning values based on the distribution of geometric distances for pairs of object clusters, which can actually be computed. For the background-background case, the average geometric distance is used as we do not want this type of merge to be penalized. The object-

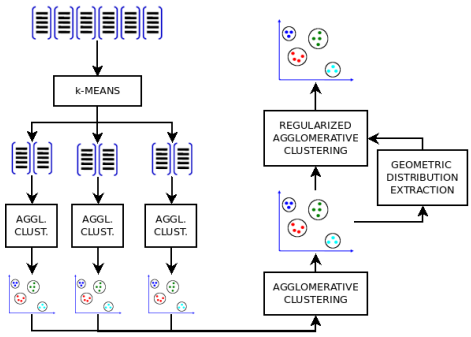


Figure 2. Regularized CPA clustering scheme.

background case, however, needs to be penalized as it would otherwise decrease cluster precision. In our experiments, we assigned in this case a value equal to the mean geometric distance between any object-object pair of clusters plus three standard deviations. We apply the heuristic that merging an object cluster with a backgorund one is as bad as merging a very different pair of object clusters. Figure 2 illustrates the procedure. We tested two measures of distance between a pair of geometric distributions: The χ^2 distance between the two histograms, and another metric based on the Kullback-Leibler divergence.

Histogram Measuring Geometric space is divided into small bins. Let H_x, H_y be two histograms for geometric distributions associated with cluster X and Y. Chi-square distance between H_x and H_y is used:

$$D_{\chi^2}(H_x, H_y) = \sum_i \frac{(H_x^i - H_y^i)^2}{H_x^i + H_y^i} \tag{3}$$

Kullback-Leibler Divergence Let P, Q be two continuous distributions, the discrete form of Kullback-Leibler divergence from P to Q, using a set of x_i point estimates, is:

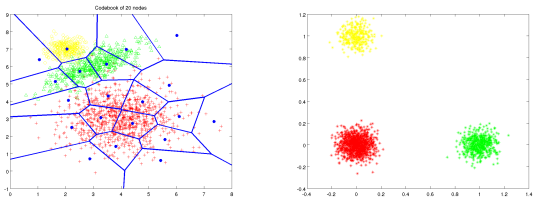
$$D(P|Q) = \sum_i p(x_i) \log \frac{p(x_i)}{q(x_i)} \tag{4}$$

which is computed using the kNN framework, as proposed in [8]. As $D(P|Q)$ is not symmetric, to make KL divergence a metric, we define a new distance which is the symmetrized KL divergence:

$$D_{SKL}(P, Q) = D(P|Q) + D(Q|P)$$

4. Experimental Results

In this section we explain the experiments done in order to demonstrate the feasibility of the proposed method. For this purpose, we designed a toy problem as a proof of concept, consisting of three *semantic classes* (Fig. 3). Although appearance clusters have some overlap and different sizes and extent, geometric clusters provide more information in



(a) (b)

Figure 3. Synthetic dataset for the toy problem. The colors represent three different semantic classes. (a) Appearance distribution with Voronoi diagram for $T_{reg} = 20$ clusters. (b) Geometric distribution.

order to better discriminate between good and bad merges, so we should rely on that to guide the clustering. With no regularization, the clustering makes a very bad merge at 12 clusters, decreasing the cluster precision, so we should have stopped. Using $\lambda = 1$, this point is at 8 clusters. Finally, with $\lambda = 8$, we can reach 6 clusters without losing cluster precision. This experiment shows that our regularization method produces more stable results in terms of preserved spatial information. For more details, the reader is referenced to [11].

For the rest of our experiments, we have used the TUD Motorbikes dataset [5]. The test set contains 115 images collected from the World Wide Web. Each image contains one or more motorbikes at different scales, usually partially occluded and in front of difficult backgrounds. We used the same training set as in [5], which consists of 153 images with uniform background of motorbike side views. The computation of geometric distribution distances is very demanding computationally. Given the advances in parallel computation techniques like CUDA technology and current GPUs, we can run algorithms with speedups superior to 100x. Our implementation took only 5 hours to cluster $260,000$ SIFT features, while a CPU-only implementation would need hundreds of hours, making it unfeasible to use. In order to see how the geometric distributions evolve as the clustering progresses, we computed the average entropy for the clusters each N iterations. First, the entropy rises because clusters are in formation. After a plateau, where we suppose that mostly background clusters are being merged, the entropy starts decreasing very fast, which gives evidence of the structure that is being consolidated, as many similar object clusters are being merged. Finally, we see that the entropy rises until we finish, suggesting that the clustering process should have stopped. After some initial test to determine the interesting parameter ranges, we used $T_{reg} = 10,000$ clusters, the regularization parameter $\lambda \in \{1, 2, 4, 8, 16, 32, 64\}$ and codebook sizes of 500, 1000, 2000, 4000 and 7000 visual words. The GHT bin size was also optimized by cross-validation. We used the overlap ratio between predicted and ground truth bounding boxes, which should be at least 50% to accept an hypothesis. As we did not implement any of the common robust hypotheses validation and refining algorithms common in the literature, we mainly focused on the impact that our method has on improving recall, and analyzed precision separately. Besides, the intention is to show how the codebook size affects false positives and true positives.

Figure 4 shows the recall results for different codebook sizes. As can be seen, using a codebook of 7000 clusters the results are virtually the same, except for high regularization factors. This was expected given the reduced range of appearance distances

compared to later stages of clustering, e.g. with 500 or 1000 clusters. More interestingly, when codebooks are smaller, from 4000 to 500 clusters, we can see how our proposed regularization scheme clearly improves the results, with up to 9% higher recall with the 500 clusters codebook. Using high regularization factors help maintaining good recall for 1000-2000 cluster codebooks. Intermediate regularization factors have a more stable behavior and obtain good results with only 500 clusters. On the other hand, the increased aperture in appearance space affects precision results when using the regularized codebook – specially for smaller codebook sizes – as shown in Figure 5. Due to time constraints, resolving this drawback by incorporating an state of the art hypothesis verification stage is left for future work.

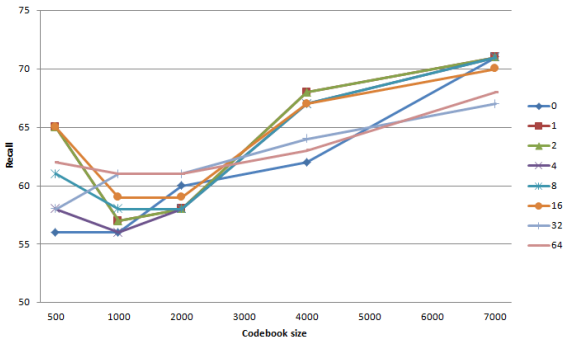


Figure 4. Recall results with different vocabulary sizes. Each line represents a regularization factor.

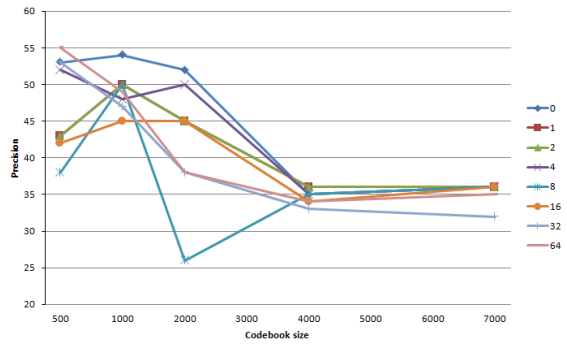


Figure 5. Precision results with different vocabulary sizes. Each line represents a regularization factor.

5. Conclusion and Future Work

In this work, we contribute a novel self-supervised radius-based codebook construction method which uses visual word spatial information to regularize the agglomerative clustering process, which allows to retain more information in the geometric space than the

standard agglomerative clustering methods. This geometric information is used to hypothesize where objects are in novel images. We are looking forward to implement the whole object recognition framework in CUDA, as it is highly parallelizable and we expect to obtain serious speedups.

The experimental results with the motorbikes dataset are quite promising, as we obtained a significantly better recall than using a non-regularized codebook. It would be interesting to automatically adjust the regularization factor to the vocabulary size, as small vocabularies typically achieved better performance and stability with higher regularization factors.

Finally, we plan to make more thorough experimentation with intermediate vocabulary sizes. Also, we will make use of a state of the art object localization framework, using our regularized codebooks there to see if those methods also benefit as much as ours.

References

[1] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.

[2] B. Fulkerson, A. Vedaldi, and S. Soatto. Localizing objects with smart dictionaries. In *10th European Conf. on Comp. Vis.*, pages 179–192, 2008.

[3] B. Leibe, K. Mikolajczyk, and B. Schiele. Efficient clustering and matching for object class recognition. In *Proceedings of British Machine Vision Conference*, 2006.

[4] B. Leibe, A. Ettlin, and B. Schiele. Learning semantic object parts for object categorization. *Image Vision Comput.*, 26(1):15–26, 2008.

[5] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *Int. J. Comput. Vision*, 77(1-3):259–289, 2008.

[6] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV '99: Proceedings of the Int. Conf. on Computer Vision-Volume 2*, page 1150, Washington, DC, USA, 1999. IEEE Computer Society.

[7] E. Murphy-Chutorian and J. Triesch. Shared features for scalable appearance-based object recognition. In *WACV-MOTION '05: 7th IEEE Workshops on Application of Comput. Vision - Vol 1*, pages 16–21, Washington, DC, USA, 2005. IEEE Computer Society.

[8] F. Perez-Cruz. Kullback-leibler divergence estimation of continuous distributions. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, pages 1666 –1670, july 2008.

[9] F. Perronnin. Universal and adapted vocabularies for generic visual categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(7):1243–1256, 2008.

[10] T. Quack, V. Ferrari, B. Leibe, and L. Van Gool. Efficient mining of frequent and distinctive feature configurations. In *Computer Vision, 2007. ICCV 2007. IEEE 11th Int, Conf, on*, pages 1–8, 2007.

[11] A. Ribes. Effective object localization with regularized codebook construction. Master's thesis, IIIA-CSIC (Univ. Autonoma de Barcelona), 2010.

[12] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. Van Gool. Towards multi-view object class detection. volume 2, pages 1589 – 1596, 2006.

[13] N. Tishby, F. Pereira, and W. Bialek. The information bottleneck method. In *37th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999.

[14] K. Torkkola. Feature extraction by non parametric mutual information maximization. *J. Mach. Learn. Res.*, 3:1415–1438, 2003.

[15] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 29:854–869, 2007.

[16] J. C. van Gemert, C. G. M. Snoek, C. J. Veenman, A. W. M. Smeulders, and J-M. Geusebroek. Comparing compact codebooks for visual categorization. *Comp. Vis. Image Underst.*, 114(4):450–462, 2010.

[17] J. Vogel and B. Schiele. Semantic modeling of natural scenes for content-based image retrieval. *Int. J. Comput. Vision*, 72(2):133–157, 2007.

[18] Y-T. Zheng, M. Zhao, S-Y. Neo, T-S. Chua, and Qi Tian. Visual synset: Towards a higher-level visual representation. *IEEE Conf on Computer Vision and Pattern Recognition*, 0:1–8, 2008.

An inference model for analyzing termination conditions of Evolutionary Algorithms

David Roche [a,1], Debora Gil [b] and Jesus Giraldo [a]
[a] *Laboratory of Systems Pharmacology and Bioinformatics, UAB, Bellaterra,Spain*
[b] *Computer Science Department and Computer Vision Center (CVC), UAB, Bellaterra,Spain*

Abstract

In real-world problems, it is mandatory to design a termination condition for Evolutionary Algorithms (EAs) ensuring stabilization close to the unknown optimum. Distribution-based quantities are good candidates as far as suitable parameters are used. A main limitation for application to real-world problems is that such parameters strongly depend on the topology of the objective function, as well as the EA paradigm used.

We claim that the termination problem would be fully solved if we had a model measuring to what extent a distribution-based quantity asymptotically behaves like the solution accuracy. We present a regression-prediction model that relates any two given quantities and reports if they can be statistically swapped as termination conditions. Our framework is applied to two issues. First, exploring if the parameters involved in the computation of distribution-based quantities influence their asymptotic behavior. Second, to what extent existing distribution-based quantities can be asymptotically exchanged for the accuracy of the EA solution.

Keywords. Evolutionary Computation Convergence, Termination Conditions, Statistical Inference

Introduction

Evolutionary Algorithms (EAs) are a class of stochastic optimization methods that simulate the process of natural evolution [1]. EAs maintain a population of possible solutions that evolve according to rules of selection and other operators, such as recombination and mutation. By their ability to optimizing non-analytic multi-modal functions, EAs have been successfully applied to a wide range of real life problems [2,3,4].

As any iterative technique, EA requires a stop criterion. Unlike optimization methods evolving a single initial value (which rely on real analysis theory), by their stochastic nature, there is not a solid mathematical theory ensuring convergence of evolutionary methodologies in general [5,1]. The simplest (and most extended [5,6,7]) stopping criterion consists in reaching a number of iterations or function evaluations. This stopping

[1]Laboratory of Systems Pharmacology and Bioinformatics, Institut de Neurociències and Unitat de Bioestadística, UAB, Bellaterra,Spain; E-mail: david.roche@uab.es.

criterion is not useful by itself (the number of iterations that guarantee convergence significantly varies across problems [5]), though it can be necessary when used in addition with alternative criteria to ensure that the algorithm stops [8].

Existing approaches defining alternative termination conditions [9,10] address the definition of a quantity reflecting the amount of change between consecutive iterations and the condition that such quantity should fulfill. Two different termination conditions are considered in the literature. Terminate either if the measure of the amount of change is below a given threshold or in the case that such measure is below a threshold for a number of generations. Concerning the amount of change, among the quantities reported in the literature [8,5], distribution-based are the best suited. These quantities measure the distribution of a given percentage of the (best) evolving population. They compare to the accuracy of the solution (distance to the optimum) in terms of number of function executions, as far as suitable parameters are set [8].

Two different sets of parameters are involved in distribution-based termination conditions. On one hand, the parameters used for the termination condition itself: threshold and number of generations. On the other hand, those involved in the computation of the distribution quantity: percentage of the population used and whether individuals are selected among the best ones or randomly. A main limitation for application to real-world problems is that the above parameters strongly depend on the topology of the objective function and the EA paradigm used [8].

The goal of this work is twofold. Firstly, determining if the parameters involved in the computation of distribution-based quantities influence the behavior of the termination condition. Secondly, exploring if the asymptotic behavior of distribution-based quantities relates to the accuracy of the EA solution. We propose posing the termination problem in statistical inference terms and introduces a general regression-prediction model for determining if two quantities behave equally. We use our model to compare several types of distribution-based quantities reported in the literature [8]. Our experiments conclude that the distribution-based parameters (percentage and selection) do not influence their asymptotic behavior and indicate that the maximum distance to the best individual might be the best choice in terms of computational efficiency and capability of predicting EA accuracy.

This paper is organized as follows. Section 1 describes the existent distribution-based termination conditions for EA algorithms. In section 2 we give the statiscal inference framework. In section 3 we describe the experiments carried out and in section 4 we report the statiscal results. Finally section 5 concludes the paper.

1. Distribution-based Stopping Criteria

A termination condition is given by a quantity reflecting the amount of change between consecutive iterations and the conditon that such quantity should fullfill. Regarding the quantity reflecting the amount of change we have considered distributed-based quantities. Such quantities measure the distribution of a given percentage of the (best) evolving population. Following the literature [8], we have considered:

1. **Maximum Distance (MxD).** It is given by the maximum distance of the population to the best individual.

2. **Population Variability (Std).** It is the maximum standard deviation of the population.

Both quantities can be computed using all individuals or considering only a percentage p of the individuals. These percentage of the individuals can be randomly sampled over the whole population or selected among the best individuals. Given these definitions, for each distribution-based quantity, two different computational procedures can be considered:

1. Percentage p over randomly sampled individuals. It will be indicated by the suffix $Prop_p$.
2. Percentage p over the best individuals. It will be be indicated by the suffix $BestProp_p$. We note that, in this case, the population must be sorted before computing the measure.

We note that the combination of the distribution-based quantities with the two computational procedures described above give four generic alternative quantities: $MaxDProp_p$, $StdProp_p$, $MaxDBestProp_p$, $StdBestProp_p$. From now on, we will note by AltCrit any of these quantities.

Concerning termination conditions, we have the following two:

1. **Absolut Threshold**. The algorithm terminates if the distribution-based quantity is below a given threshold. This condition requires knowing the expected ranges of the distribution-based quantity in order to set the appropriate threshold. Besides it does not guarantee that EA has reached a local minimum.
2. **Stabilization**. The algorithm terminates if the distribution-based quantity is below a given threshold for a number of consecutive generations. This condition is equivalent to have differences between consecutive iterations below a given threshold. On one hand, the latter threshold is independent of the ranges of the distribution-based quantity. On the other hand, this condition ensures that the distribution-based quantity has stabilized and, thus, that EA does not evolve anymore since a local minimum has been attained.

A termination condition is suitable if it compares to the accuracy of the EA solution. The distance to the (known) function minimum is our gold-standard reference convergence criterion, given that is directly associated to the algorithm accuracy. This criterion can only be computed if the optimum of the test function is known and, thus, is useless in real-world problems. We compute it as the maximum distance to the function minimum of a certain percentage p of the individuals [8] and note it by $RefCrit$.

Our final goal is to control (predict) the values taken by $RefCrit$ from the values taken by the alternative measure $AltCrit$. Previous to the latter, we should analyze the influence that the different computational procedures may have on the distribution-based measures. In inference statistics, this can be achieved by relating quantities using a regression model.

2. Inference Model

Given a sampling of two random variables (x and y), the linear regression of y (response variable) over x (explicative variable) is formulated as:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \tag{1}$$

for x_i, y_i the sampling of x and y and ε_i an uncorrelated random error following a multivariate normal distribution, $N(0, \Sigma^2)$ of zero mean and variance $\Sigma^2 = \sigma^2 Id$.

The parameters of the regression model (1) are the regression coefficients $\beta = (\beta_0, \beta_1)$ and the error variance σ^2. The regression coefficients describe the way the two variables relate, while the variance indicates the accuracy of the model and, thus, measures to what extent x can predict y.

In the case of exploring equivalence among $AltCrit$, the inference model would be:

$$AltCrit1_i = \beta_0 + \beta_1 AltCrit2_i + \varepsilon_i \tag{2}$$

for $AltCrit1_i$, $AltCrit2_i$ the values of two different $AltCrit$ obtained at the i-th iteration. In the case of relating $AltCrit$ to $RefCrit$, our model would be:

$$RefCrit_i = \beta_0 + \beta_1 AltCrit_i + \varepsilon_i \tag{3}$$

for $RefCrit_i$, $AltCrit_i$ the values of $RefCrit$ and $AltCrit$ obtained at the i-th iteration.

For a sample of length N, the regression coefficients, $\widehat{\beta} = (\widehat{\beta}_0, \widehat{\beta}_1)$, are computed by Least Squares Estimation (LSE) as:

$$\widehat{\beta} = (X^T X)^{-1} X^T Y \tag{4}$$

for $X = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_N \end{pmatrix}$, $Y = (y_1, \ldots, y_N)$ and ^{T} denoting the transpose of a matrix. The differences between estimated responses, $\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$, and observed responses y_i:

$$e_i = y_i - \widehat{y}_i \tag{5}$$

are called residuals. Their square sum provides an estimation of the error variance:

$$S_R = \widehat{\sigma}^2 = \frac{\sum e_i^2}{n-2} \tag{6}$$

Due to a decrease in the population sparseness at advanced stages of EA, the heterocedasticity and Gaussianity assumption are not satisfied. A monotonous increase in σ^2 is usually solved by taking logarithms in both variables [11].

From now on, the values of $RefCrit$ and $AltCrit$ will be assumed to be in logarithmic scale for the inference model:

$$log(y_i) = \beta_0 + \beta_1 log(x_i) + \varepsilon_i \tag{7}$$

We note that, by taking exponentials, the regression model in the original scale is polynomial with multiplicative errors:

$$y_i = e^{\beta_0} x_i^{\beta_1} e^{\varepsilon_i} \tag{8}$$

Previous to any kind of inference, it is mandatory to verify that the estimated parameters make sense. That is, whether it really exists a linear relation between x and y. By the Gauss-Markov theorem, such linear relation can be statistically checked using the following T-test [12]

$$TM:\ H_0 : \beta_1 = 0\ , H_1 : \beta_1 \neq 0 \tag{9}$$

where a $p - value$ close to zero (below α) suggests that a linear model can be used.

We note that if the slope $\beta_1 \approx 1$, then both quantities stabilize at the same time and, thus, they are equivalent under the stabilization termination condition. This requirement can be statistically checked using the following unilateral T-tests:

$$\begin{array}{l} TP_1 : H_0 : \beta_1 - 1 \geq \epsilon \quad\ ,\ H_1 : \beta_1 - 1 < \epsilon \\ TP_2 : H_0 : \beta_1 - 1 \leq -\epsilon\ ,\ H_1 : \beta_1 - 1 > -\epsilon \end{array} \tag{10}$$

a p value close to zero (below α) for both tests ensures that $|\beta_1 - 1| \leq \epsilon$ with a confidence $(1 - \alpha)100\%$. The minimum ϵ ensuring rejection of TP_1 and TP_2 is given by $|1 - max(|CI(\beta_1)|)|$ for $CI(\beta_1)$ the $(1-\alpha)$-confidence interval for the regression slope. The higher ϵ we have, the least stabilization equivalence.

3. Experimental Settings

In order to compare the different distribution-based quantities, we have considered six well-known test functions [13] having a minimum at zero:

1. Esphere:

$$f_1(x) = \sum_{i=1}^{n} x_i^2$$

2. Rosenbrock:

$$f_2(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$$

3. Rastrigin:

$$f_3(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i + 10)]$$

4. Griewangk:

$$f_4(x) = \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + 1$$

5. Ackley:

$$f_5(x) = 20 + e - 20e^{-0.2\sqrt{\frac{1}{2}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{2}\sum_{i=1}^{n} cos(2\pi x_i)}$$

6. Easom:

$$f_7(x) = -\cos x_1 . \cos x_2 . \\ \exp(-((x_1 - \pi)^2 + (x_1 - \pi)^2))$$

We have used a Differential Evolution (DE) technique for the minimization task. Differential evolution is a real parameter encoding evolutionary algorithm for global optimization over continuous spaces [14,15]. In this paper, we use the 3-parameter DE1 scheme [14] for solving DE. For a real search space of dimension D, the population is randomly initialized with ND vectors (for ND the first algorithm parameter). Each vector v in the population is evolved by mutation and recombination operators. Given a mutation rate $F \in [0, 2]$ (second parameter of the algorithm), the mutation operator produces a new vector vm by adding a vector difference of two randomly chosen population vectors $v1$ and $v2$ to another randomly chosen vector $v3$:

$$vm = v1 + F(v2 - v3) \tag{11}$$

For the recombination step, a new vector vf is created from the mutation vector by means of a combination rate CR (third parameter of the algorithm) as follows:

$$vf_i = \begin{cases} vm_i & \text{if} \quad r_i < CR \text{ or } i = k \\ v_i & \text{otherwise} \end{cases} \tag{12}$$

for vf_i the i-th component of vf and $r_i \in [0, 1]$ a random number and k a random number uniformly distributed in $[1, D]$. Finally a selection operator is applied. The vector vf and the initial vector v are compared and the vector that better fits the objective function is selected and remains in the next population. This process is iteratively repeated until a stopping criterion is reached. Following the literature [15], we have chosen the following values for DE parameters: D=2, ND=20, F=0.9, CR=0.5. For each test function, we have executed 100 trails of the algorithm during 10.000 iterations each one.

Two different experiments have been carried out:

1. **Computation of Distribution-based Quantities.** A first compulsory step before exploring if any $AltCrit$ can substitute $RefCrit$ is checking if the computation of distribution-based quantities using different parameter settings yields equivalent quantities. In the case such equivalency held, it would imply that **MxD** and **Std** can be computed directly from EA current population. Otherwise, EA output should be adapted in order to conform to the most adequate proportion and selection strategy.
2. **Capability for Substituting the Reference Criterion.** Under the assumption that any parameters can be used for computing $AltCrit$, we can compare it to $RefCrit$ using the natural setting for DE: a percentage ($p = 30\%$) of the whole population.

For both experiments, we have assessed the validity of the linear model in logarithmic scale (given by S_R and TM test), as well as its stabilization equivalence (given by the ϵ rejecting TP_i, $i = 1, 2$).

4. Experiments and Results

4.1. Computation of Distribution-based Quantities

Tables 1 and 2 report the estimation of the model parameters (the regression coefficients $\widehat{\beta}_0$, $\widehat{\beta}_1$ and the residual variance S_R), the p-value of the model verification T-test and the stabilization equivalence tolerance ϵ. Table 1 shows results for the comparison between best and randomly selected individuals for the proportion $p = 60\%$ and the two distribution-based quantities. Table 2 shows results for the comparison across proportions for **MxD**.

For all cases, there is a clear linear relation between the different quantities (with p close to the working precision). We can assume that $\widehat{\beta}_1$ is close to 1 with a tolerance $\epsilon \approx 10^{-2}$ for dependency on the best individuals (table 1) and $\epsilon \approx 10^{-1}$ for dependency on proportions (table 2). Therefore, all computational strategies are equivalent for the stabilization termination condition. Besides, the sign of the constant term $\widehat{\beta}_0$ gives the following (expected) inequalities. For any percentage p, we have:

$$\mathbf{MxDBestProp}_p < \mathbf{MxDProp}_p \text{ and } \mathbf{StdBestProp}_p < \mathbf{StdProp_p}$$

and for increasing proportions:

$$\mathbf{MxDProp_{30}} < \mathbf{MxDProp_{60}} \sim \mathbf{MxDProp_{100}}$$

The above inequalities indicate that best individuals are a more compact cluster and that the estimation of the population dispersion increases with the number of samples considered.

	MxD					**Std**				
	TM	$\widehat{\beta}_1$	$\widehat{\beta}_0$	S_R	ϵ	TM	$\widehat{\beta}_1$	$\widehat{\beta}_0$	S_R	ϵ
Esphere	$\leq 10^{-30}$	0.99	0.75	0.11	0.00002	$\leq 10^{-30}$	1.00	0.62	0.07	0.00005
Rosenbrock	$\leq 10^{-30}$	0.99	0.51	0.14	0.00274	$\leq 10^{-30}$	0.99	0.43	0.09	0.00259
Rastrigin	$\leq 10^{-30}$	1.03	0.80	0.10	0.04029	$\leq 10^{-30}$	1.03	0.74	0.08	0.03576
Griewangk	$\leq 10^{-30}$	1.03	0.62	0.33	0.03243	$\leq 10^{-30}$	1.02	0.59	0.27	0.02988
Ackley	$\leq 10^{-30}$	1.02	0.99	0.10	0.02530	$\leq 10^{-30}$	1.02	0.84	0.07	0.02094
Easom	$\leq 10^{-30}$	1.05	1.06	0.05	0.05647	$\leq 10^{-30}$	1.04	0.90	0.03	0.04537

Table 1. Comparison between best or random selection for a percentage $p = 60\%$ of the population.

4.2. Capability for Substituting the Reference Criterion

Given that the setting used for the computation of alternative quantities is irrelevant for termination by stabilization, we have computed **MxD** and **Std** for a uniform sampling of $p = 30\%$ of the population. Table 3 reports the estimation of the model parameters (the regression coefficients $\widehat{\beta}_0$, $\widehat{\beta}_1$ and the residual variance S_R), the p-value of the model verification T-test and the stabilization equivalence tolerance ϵ. We report values for each test function (rows) and alternative quantity (columns). For all cases, there is a clear lin-

	Proportion p=30% versus p=60%					**Proportion p=30% versus p=100%**				
	TM	$\widehat{\beta}_1$	$\widehat{\beta}_0$	S_R	ϵ	TM	$\widehat{\beta}_1$	$\widehat{\beta}_0$	S_R	ϵ
Esphere	$\leq 10^{-30}$	0.99	-0.5	6.32	0.00081	$\leq 10^{-30}$	1.00	-0.6	5.23	0.00344
Rosenbrock	$\leq 10^{-30}$	0.97	-0.4	4.85	0.01825	$\leq 10^{-30}$	0.96	-0.4	5.36	0.03187
Rastrigin	$\leq 10^{-30}$	0.99	-0.2	0.48	0.00804	$\leq 10^{-30}$	0.98	-0.6	0.42	0.02287
Griewangk	$\leq 10^{-30}$	0.95	-1.0	3.36	0.04759	$\leq 10^{-30}$	0.80	-3.4	11.75	0.19316
Ackley	$\leq 10^{-30}$	0.99	-0.1	0.63	0.00212	$\leq 10^{-30}$	0.99	-0.2	0.53	0.00113
Easom	$\leq 10^{-30}$	0.99	-0.3	0.40	0.01058	$\leq 10^{-30}$	0.99	-0.3	0.48	0.00611

Table 2. Comparison across percentages for individuals randomly chosen in the computation of **MxD**.

	MxDProp$_{30}$					**StdProp**$_{30}$				
	TM	$\widehat{\beta}_1$	$\widehat{\beta}_0$	S_R	ϵ	TM	$\widehat{\beta}_1$	$\widehat{\beta}_0$	S_R	ϵ
Esfere	$\leq 10^{-32}$	1.000	-0.183	0.05	0.000	$\leq 10^{-32}$	1.000	0.685	0.03	0.000
Rosenbrock	$\leq 10^{-32}$	1.002	-0.122	0.09	0.002	$\leq 10^{-32}$	1.003	0.761	0.05	0.003
Rastrigin	$\leq 10^{-32}$	1.006	-0.207	0.06	0.006	$\leq 10^{-32}$	1.007	0.686	0.03	0.006
Griewangk	$\leq 10^{-32}$	1.009	-0.1714	0.02	0.007	$\leq 10^{-32}$	1.008	0.7021	0.01	0.007
Ackley	$\leq 10^{-32}$	1.004	-0.148	0.06	0.004	$\leq 10^{-32}$	1.004	0.725	0.03	0.004
Easom	$\leq 10^{-32}$	1.011	-0.143	0.04	0.011	$\leq 10^{-32}$	1.011	0.766	0.03	0.011

Table 3. Comparison to $RefCrit$.

ear relation between accuracy and the alternative quantities (with p close to the working precision). Besides the goodness-of-fit is excellent, given that S_R is extremely small compared to the variable ranges.

Concerning the relation between the two variables, it is worth noticing two aspects. Firstly, we observe that the estimated slope $\widehat{\beta}_1$ is close to 1 for all cases with tolerance $\epsilon \leq 0.11$ for both quantities **MxD** and **Std**. This implies that the relation in logarithmic scale is a translation of the identity and the regression model in the original scale is also linear. Secondly, the constant coefficients $\widehat{\beta}_0$ are sorted as follows:

$$\widehat{\beta}_0(\mathbf{MxDProp}) \leq 0 \leq \widehat{\beta}_0(\mathbf{StdProp})$$

The above commented points indicate that there might be the following tendency:

$$\mathbf{StdProp} \leq RefCrit \leq \mathbf{MxDProp}$$

This already suggests that the value of maximum distances itself might guarantee an upper bound for the EA accuracy.

5. Conclusions and Future work

A main challenge in Evolutionary Algorithms (EAs) is determining a termination condition ensuring stabilization close to the optimum in real-world applications. Although distribution-based conditions are the best suited, a major concern is setting appropriate parameters for their computation. In this context, this paper addresses EA termination

condition in terms of statistical inference and contributes in two issues. Firstly, it explores if the parameters involved in the computation of distribution-based quantities influence the behavior of the termination condition. Secondly, it reports a preliminary study on the relation between such termination conditions and the accuracy of the EA solution.

According to our experiments on several known test functions, the following conclusions can be derived. On one hand, there is a high correlation among the different computational procedures for distribution-based quantities and they behave equally with respect their stabilization. Therefore, we conclude that in order to use a quantity behaving as the solution accuracy we can choose, among the various computational procedures, the most convenient for the particular EA paradigm we are running. On the other hand, there is a strong linear relation between distribution-based quantities and the distance to the optimum. From our analysis, we conclude that quantities based on maximum distances have the highest concordance to EA accuracy, in the sense that they guarantee an upper bound for the accuracy. Thus, they are the best candidates for terminating EA in real-world problems.

We consider that there are some issues that should be further developed. The test functions used are a small set of benchmarking data sets (we cover two out of the five categories described in [16]) and only 2-D problems have been solved. Enlarging the test function data set including groups of functions with specific key features [16] is work currently under development. However, the functions used include three properties (multimodality, global structure and scalability) reported in a recent study [17] to have a high influence in the performance of EA's. Regarding size, although it definitely influences convergence rate (more iterations of EA are required [16]), this is independent of the relationship between *RefCrit* and *AltCrit*. Thus, size is not a limitation for the prediction model, which links convergence rate with population stability.

In this study we have restricted to DE algorithm. We are currently enlarging EA methods in order to cover existing EA paradigms: genetic algorithms [18], evolutionary strategies [19], particle Swarm optimization [20], among others. Nevertheless, we do not expect any significant changes in our conclusions since DE already presents the main features of EA [21].

Finally, in our experimental setting the test functions have been studied separately. We consider that the influence of the test function should be taken into account, so that the inference can be done independently of the function features. This will be studied by using generalized regression models including random effects [22] modelling the impact of the test function group.

Acknowledgements. This work was supported by the Spanish projects PI071188, TIN2009-13618, CONSOLIDER-INGENIO 2010 (CSD2007-00018), SAF2010-19257 and Fundació La Marató de TV3 (070530). The 2nd author has been supported by The Ramon y Cajal Program.

We would like to thank Llorenç Badiella from the Statistics Service at the Universitat Autonoma de Barcelona for his advise.

References

[1] Bäck, T., Hammel, U., Schwefel, H.: Evolutionary computation: Comments on the history and current state. IEEE Transactions on Evolutionary Computation **1** (1997) 3–17

[2] Ravikumar, V., Panigrahi, B.: Comparative study of evolutionary computing methods for parameter estimation of quality signals. International Journal of Applied Evolutionary Computation **1 (2)** (2010) 28–59
[3] Rizki, M., Zmuda, M., Tamburino, L.: Evolving pattern recognition systems. IEEE Transactions on Evolutionary Computation **6 (6)** (2002) 594–609
[4] Cagnoni, S., Lutton, E., Olague, G.: Genetic and Evolutionary Computation for Image Processing and Analysis. Hindawi Publishing Corporation (2008)
[5] Safe, M., Carballido, J., Ponzoni, I.: On stopping criteria for genetic algorithms. In: 17th BSAI, LNAI 3117. (2004) 405–413
[6] Price, K., Storm, R., Lampinen, J.: Differential Evolution: A Practical Approach to Global Optimization. Springer (2005)
[7] Tagetiren, M., Suganthan, P.: A multi-populated differential evolution algorithm for solving contrained optimization problem. In: IEEE Congres on Evolutionary Computation. (2006) 340–347
[8] Zielinski, K., Laur, R.: Stopping criteria for differential evolution in constrained single-objective optimization. In: Advances in Differential Evololution, vol. 143. (2010) 111–138
[9] Zielinski, K., Laur, R.: Stopping criteria for a constrained single objective particle swarm optimization algorithm. Informatica 31(1) (2007) 51–59
[10] K. Zielinski, D.P., Laur, R.: Stopping criteria for single-objective optimization. In: Third International Conference on Computational Intelligence, Robotics and Autonomous Systems, Singapore. (2005)
[11] Arnold, S.: The theory of linear models and multivariate observations. Wiley (1997)
[12] Newbold, P., Carlson, W., Thorne, B.: Statistics for Business and Economics. 6th edn. Pearson Education (2007)
[13] Digalakis, J., Margaritis, K.: An experimental study of benchmarking functions for genetic algorithms. International Journal of Computer Mathematics **79(4)** (2002) 403–416
[14] Storn, R., Price, K.: Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization **11** (1997) 341–359
[15] Das, S., Konar, A.: An improved differential evolution scheme for noisy optimization problems. In: LNCS vol. 3776. (2005) 417–421
[16] Hansen, N., Auger, A., Ros, R., Finck, S., Posík, P.: Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009. In: GECCO 2010. (2010)
[17] Mersmann, O., Preuss, M., Trautmann, H.: Benchmarking evolutionary algorithms: Towards exploratory landscape analysis. In: PPSN XI, Part I, LNCS 6238. (2010) 73–82
[18] Goldberg, D., Richardson, J.: Genetic algorithms with sharing for multimodal function optimization. In: Second International Conference on Genetic Algorithms and their Application. (1987) 41–49
[19] Beyer, H.G., Schwefel, H.P.: Evolution strategies: A comprehensive introduction. Journal Natural Computing **1(1)** (2002) 3–52
[20] Barrera, J., Coello, C.A.C. In: A Review of Particle Swarm Optimization Methods used for Multimodal Optimization. Springer-Verlag (2009) 9–37
[21] Ronkkonen, J.: Continuous Multimodal Global Optimization with Diferential Evolution Based Methods. PhD thesis, Lappeenranta University of Technology (2009)
[22] Lee, Y., Nelder, J., Pawitan, Y.: Generalized linear models with random effects. Chapman and Hall/CRC Boca Raton (2006)

Gait recognition by using Spectrum Analysis on state space reconstruction

Albert SAMÀ[a,1] Francisco J. RUIZ[a], Carlos PÉREZ-LÓPEZ[a] and Andreu CATALÀ[a]
[a] *Universitat Politècnica de Catalunya*

Abstract. This paper describes a method for identifying a person while walking by means of a triaxial accelerometer attached to the waist. Human gait is considered as a dynamical system whose attractor is reconstructed by time delay vectors. A Spectral Analysis on the state space reconstruction is used to characterize the attractor. The method is compared to other common methods used in gait recognition tasks through a preliminary test.

Keywords. Gait recognition, Spectral methods, Inertial sensors

Introduction

Human movement analysis is a research field with clinical and biometrics application. It has been shown useful in the objective measurement of gait, balance, falls risk assessment and mobility monitoring [1]. Biometric identification is also a field of great interest whose research covers security and access control applications. Typical identification systems analyze fingerprints, speech or iris. Recent studies try to perform it by more complex patterns like those obtained by gait [2]. The existing gait recognition methods can be grouped into three categories: vision based, floor sensor based and inertial sensor based. In this work, we focus our study in the third category.

Previous studies on biometric identification based on user's gait employed a mobile phone [4, 5] or a specifically developed device with an accelerometer [3,6]. The methodology used by those studies consists in segmenting the accelerometer signal in gait cycles and characterizing each cycle by some features (e.g. cumulants [4], histograms and correlations [3, 5 and 6]).

Dynamical systems give us a different approach to analyze human movement. It is based on Taken's theorem; thus sensor measures are treated as time series in order to reconstruct the attractor of the dynamical system being sensed. Then, reconstructed space is characterized by a spectral analysis. This approach is followed by this work, and it has been tested previously for extracting step length and velocity by using a triaxial accelerometer [8]. Similar methods have been previously used for human full-body pose tracking by means of six inertial sensors [9] and for activity classification [10]. The greatest advantage of the method over other methods is its ability to characterize an activity by using only a few number of features and regardless the orientation of the sensor.

[1] Corresponding Author: Albert Samà, Neàpolis Building, Rambla de l'Exposició, 59-69 08800 Vilanova i la Geltrú, Spain; E-mail: albert.sama@upc.edu

The paper is organized as follows: in the next Section, common gait recognition methods are described. Next, a brief introduction to the theory of state space reconstruction and spectrum analysis and some remarks on practical implementation are presented. Section 3 is devoted to introduce the approach used in this work, which is based on applying the described spectral method to the problem of gait identification. Experiments and the analysis of the results are described in section 4. Finally, section 5 includes the conclusion and future research issues.

1. Time-domain gait recognition methods

Most common gait recognition methods perform the identification of a person by using gait cycles [3, 4, 5, 6], i.e. one stride or two steps. For each user, a representative gait cycle must be constructed in a training phase. Then, the recognition process consists in comparing a new stride to the representative cycle of each user. The new stride is assigned to the most similar representative cycle, which provides the user.

The gait recognition methods described in this section are called as time-domain methods in order to differentiate them from those presented in section 2 that uses the reconstruction of the state space.

1.1. Segmentation process

The objective of segmenting is to divide accelerometer signal in gait cycles, i.e. each stride is isolated. This process can be performed by two different methods. On the one hand, a simple minimum peak detection based on lateral acceleration has been used in [4, 5, 6]. This kind of analysis is valid only when the accelerometer is located at the waist, a different analysis would be necessary if the sensor was attached to a different position. On the other hand, autocorrelation function of vertical acceleration may be used for step detection [3]. This method would be valid for almost any position the sensor can be attached. Figure 1 shows an example of a segmentation process result by using autocorrelation function.

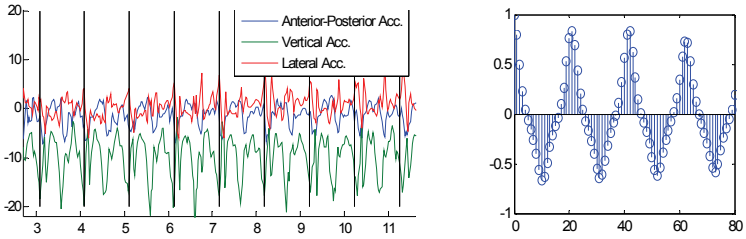


Figure 1. Gait segmentation on the left. Example of autocorrelation obtained on the right.

In this work, strides are detected by using the autocorrelation function. The minimum-peak-based detection does not perform well in every user since some gaits are irregular and produce occasional minimum peaks that result in a bad segmentation.

1.2. User representative cycle

Once cycles from training data have been extracted, it is necessary to construct a representative stride in order to compare new cycles to it. The simplest approach consists in normalizing the cycles by length and amplitude and then average them to obtain the representative stride by mean [6] or median [5]. The most common approach is to use a histogram with a fixed number of bins [3, 5, 6], where features comprise the number of values relying in each bin. Note that normalization in time and amplitude is needed. Correlation between axis [3, 5, 6] and cumulant coefficients of order 1 to 4 [5] has been also used for characterizing gait cycles. In this work, the combination of those possible features that maximize the gait recognition accuracy has been chosen.

Once representative strides for each user have been obtained, the identification would be performed in the following way. Given a signal whose user is needed to be recognized, first it is divided into strides by the method showed in the previous subsection. Then, each stride is characterized by the same features used to characterize the representative stride. The signal is assigned to the user whose representative stride is more similar to the new stride being analyzed.

2. State space reconstruction

In this Section, a brief introduction to the theory of state space reconstruction and some remarks on practical implementation is presented. State space reconstruction methods have been developed as a mean to obtain a topologically equivalent representation of the state space from one or more observed signals of a dynamical system.

2.1. Delay coordinates

A scalar time series can be considered as a one-dimensional observed measures obtained from a smooth *d*-dimensional dynamical system. The original *d*-dimensional state space of the dynamical system cannot be directly observed from the time series. However, it is possible to reconstruct this original state space or, at least, a topologically equivalent embedded space from the called *delay coordinates* [11].

Considering a single time series measured every time step Δt $\{s_{t_0}, s_{t_0+\Delta t}, \ldots\}$ (where Δt is the inverse of the sampling frequency), the *delay coordinates set* with dimension m and time lag τ is formed by the time delayed values of the scalar measurements $\mathbf{r}_t = \{s_{t-\tau(m-1)\Delta t}, \ldots, s_{t-\tau\Delta t}, s_t\} \in \mathrm{R}^m$. For notation simplicity, henceforth, time step Δt is avoided. Takens proved in 1980 the well known *Takens' embedding theorem* [12], which states that if the time series comes from a noiseless observation of a smooth dynamical system, the attractor recovered by delay coordinates is topologically equivalent to the original attractor in the state space. Even though Takens' theorem does not give guarantees of the success of the embedding procedure in the noisy case, the method has been found useful in practice.

There is a large literature of the "optimal" choice of the embedding parameters m and τ. It turns out, however, that what constitutes the optimal choice largely depends on the application [13]. In terms of the time lag τ, one of the most extended method to determine the optimal delay time was suggested by Fraser and Swinney [14]. They suggest using the first minimum in delayed average mutual information function. On

the other hand, a method to determine the minimal sufficient embedding dimension *m* was proposed by Kennel et al. [15] [16]. The idea is related to topological properties of the embedding and consists of computing the percentage of false neighbors, i.e. closer points that are no longer neighbors if the embedding dimension increases, which allows the sufficient embedding dimension to be determined.

2.2. Singular Spectrum Analysis

If Taken's theorem requirements are accomplished, the time delay coordinates leads to an embedding of the original state's space. Then, every linear transformation of sufficient rank from the time delay coordinates also leads to an embedding. A good choice of linear transformation is known as *principal component analysis* (PCA). This technique is widely used, for example to reduce multivariate data to a few dimensional data. The idea is to introduce a new set of orthonormal basis vectors in embedding space such that projections onto a given number of these directions preserve the maximal fraction of the variance of the original vectors. Solving this problem leads to an eigenvalue problem. The orthogonal eigenvectors obtained from the autocovariance matrix determine the *principal directions*. By considering only a few of this directions (those with largest eigenvalues) is sufficient to represent most part of the embedded attractor.

Singular Spectrum Analysis (SSA) [17] consists of applying a PCA, or other similar methods of spectra decomposition, to the set of delay coordinates, hereafter to be called *reconstructed attractor* or *reconstructed space*. This analysis is applied in this case as follows: given a time delayed vector $\mathbf{r}_t=(s_{t-\tau(m-1)},\ldots,s_t)$, which reconstructs the attractor for the actual state $\mathbf{x}_t$ at time t, a matrix which reconstructs the trajectory from time t to time $t+w$ is:

$$\mathbf{M}_t=[\ \mathbf{r}_t\ \mathbf{r}_{t+\tau}\ldots\mathbf{r}_{t+k\tau}\ ]^T \qquad (\mathbf{1})$$

where $k=w/\tau$.

Such matrix is first set to have zero mean (that leads to matrix $\mathbf{M}^0_t$) and then analyzed by applying a PCA process, so $\mathbf{M}^0_t$ is decomposed such that:

$$\mathbf{M}^0_t=\mathbf{U\Sigma V^*} \qquad (\mathbf{2})$$

V represents a change of basis from the reconstruction space to the latent space. Thus, $\mathbf{VM}^0_t$ identifies the trajectory of the reconstruction of the states $\mathbf{x}_t$, $\mathbf{x}_{t+1},\ldots,\mathbf{x}_{t+w}$ in the latent space.

3. Gait identification approach

This section describes the approach used in this work, which is based on applying the spectral method described in the previous section on the state space reconstruction in order to perform gait identification.

The accelerometer signal is measured at each time t as a triplet composed of three accelerations from the three axes of the sensor. Its magnitude is used as the scalar measure that will lead to reconstruct the state space:

$$s_t=(x_t^2+y_t^2+z_t^2)^{1/2} \qquad (\mathbf{3})$$

Thus, only magnitudes are going to be used which provide a method independent of the orientation.

Gait is a process of cyclic nature and its sequence of states is expected to be essentially periodic. Thus, trajectories in the state space reconstruction, i.e. the sequence of reconstructed states, should be also closed orbits and a cyclical behavior should be observed in the generated matrix $\mathbf{M}_t$. Different gaits are expected to provide different orbits, so the characterization of those orbits may allow us to identify which person belongs to. The orbit characterization from PCA is considered by two ways. Firstly, the directions where maximum variance is achieved are expected to characterize the dynamical system, as each trajectory should take a different form. Secondly, the eigenvalues are assumed to describe the transformation between latent and reconstruction space, so each transformation would be particular for each gait.

Embedding dimension m, time-lag τ and window size w are set through the characteristics of the dynamical system. Different values are tested for m and w considering both the attractor dimension and the number of states that a cycle takes. The time lag will be fixed by the results obtained by Average Mutual Information. This approach is different from classical spectral methods where an arbitrarily large value for m parameter is fixed without evaluating its effect [19].

4. Experiments

A database composed of 233 steps belonging to five users has been used for the experiments. The healthy volunteers walked 20 m at normal speed twice. A device containing a triaxial accelerometer developed at CETpD [8] and located at the lateral side of the waist logged accelerations in a sampling frequency of 200 Hz.

4.1. Time-domain gait recognition

A time-domain gait recognition method is applied to accelerometer signal in a lower sampling frequency in order to use the minimum number of samples required. Since frequency content is known to be below 20 Hz [18], signal is resampled from 200 Hz to 40 Hz.

Signal segmentation into steps is performed by detecting maxima in the autocorrelation function applied to the vertical acceleration. It has been observed that left and right steps of a user may be quite different. A classification based on steps should take into account those differences, but it has been also observed that the stride composed by both of them remain constant in a user. This observation was also remarked in [6]. Thus, the elements considered to be identified are the strides.

Once a stride has been identified, features are extracted from it. The histogram technique is applied after normalizing in time and amplitude. As in [3, 6], each stride is interpolated into 100 values and the quantity remained in the 10 bins of the histogram are used to characterize the step. Correlation between axes, kurtosis and skewness are also used, similarly to [6].

4.2. State space reconstruction

Part of the results reported in this section has been published in a recent article [7]. Figure 2 shows average mutual information (AMI) applied to the signals obtained for

each volunteer as a function of the time lag parameter. AMI measures the dependence among reconstructed states. A small value of time lag produces correlated states which may do not allow trajectories of the dynamical system appear, and a high value of time lag produces independent states, which may convert the sequence of states to a random process. However, time lag influences the attractor reconstruction by its order of magnitude but not by its specific value. A suitable value for time lag is considered to be the first minima when its value is observed against the AMI value. Observing Figure 2, a suitable time-lag value for all volunteers is 10 times the original time step. Therefore, as the original time step was 1/200 Hz = 5 ms, new time step is 50 ms and the resulting frequency sampling is 20 Hz.

As in the previous section is described, cyclical properties of gait must be shown in the reconstructed trajectories at $\mathbf{M}_t$. Recurrence plots, which are a common technique helpful to visualize the recurrences of dynamical systems, may be used with this purpose. Given a sequence of reconstructed states $x_1,..,x_n$, a matrix $\mathbf{R}_n$ is considered where each element m_{ij} may have two values: 1 when $x_i \approx x_j$ and 0 otherwise. Note that similarity is dened by ε-insensitivity. This matrix is plotted, so that recurrence plots are obtained, and periodic motions are reflected by long and non-interrupted diagonals. The vertical distance between these lines corresponds to the period of the oscillation. Figure 3 shows the recurrence plot for a volunteer when using embedding dimension 5 and 30. For the lower dimension, the periodic motion is not as clear as in the higher dimension where the cyclic motion appears obvious. This results agrees with Taken theorem.

From Figure 3 it is shown that the period of the orbit in the state space is the same for both embedding dimensions, and is reckoned to be ~30 reconstructed states. The rest of volunteers provide a similar period. FNN algorithm gives 5 as the minimum embedding dimension for all volunteers. Taking into account results from recurrence plots and FNN, *m* parameter is tested with values from 5 to 30.

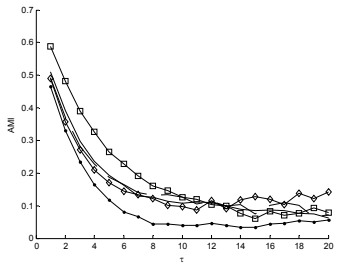


Figure 2. AMI results for all five signals. A time lag of 10 is suitable for all time series

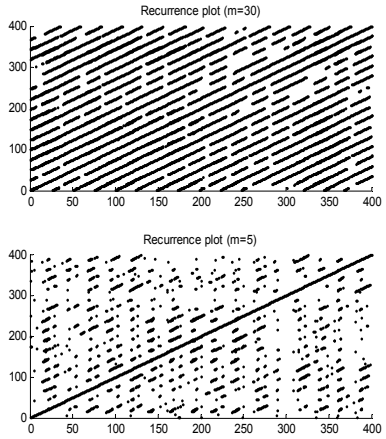


Figure 3. Recurrence plot when using embedding dimension 5 (up) and 30 (down).

Since orbits comprise 30 samples and sampling frequency was 200 Hz, a whole orbit takes $30 \cdot \tau/200=1.5$ seconds approximately. In order to test whether half a period, one or two periods enable to recognize the system, window size values used are: w=0.75 s., w=1.5 s. and w=3 s.

4.3. Results and discussion

Both time-domain gait recognition and state space reconstruction training processes are performed by Classification and Regression Trees (CART) using the first 20 m. walked by volunteers. Accuracies are obtained classifying the second 20 m. walk. CART methodology used is the standard cross-validation prune, where the optimal tree is the one with least nodes whose accuracy is within 1 standard error of the minimum cost tree.

Time-domain gait identification achieves 82% of accuracy for the 5 users. This percentage is similar to described in [5] (86%), and quite below than the percentage in [3], which was 95% although this paper located the accelerometer at the leg. Accelerations from leg show clearer the behavior of gait than those obtained when the sensor is at the waist, which may explain the higher performance.

Gait identification results using State Space Reconstruction approach are shown in figure 4. A sequence of reconstructed states, i.e. matrix $\mathbf{M}_t$, is characterized by using the eigenvalues of the matrix as features. The coefficients of the Principal Components (PC), which determine their direction, may be used but they were found to obtain poorer results [7]. Accuracies are shown as a function of the window size w, embedding dimension m and the number of eigenvalues used.

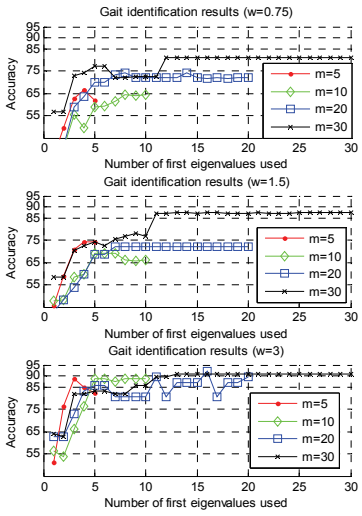


Figure 4. Gait identification results

Results show that it is possible to identify a person from a gait classifier with an overall accuracy of 90%.

When gait identification is performed by means of eigenvalues, a better identification is also obtained for larger window sizes. A window size containing half a period (w=0.75) provides a maximum accuracy of 82%, for a whole period (w=1.5) achieves 88% of accuracy, and for two-periods size (w=3) obtains the highest accuracy: 92.3%. Best classification results are obtained for m=30 when window sizes are w=0.75 and w=1.5. For w=3 such value of m also provides the highest accuracy, but m=20 does as well. Thus, it can be concluded that, in general, unfolding whole orbits provide good results.

However, it may be used lower embedding dimensions for some windows sizes, which would save computational costs. Observing the number of eigenvalues needed to achieve the maximum accuracy for m=30, it may be seen that less than a half of them is

needed. Thus, the embedding approach used is able to characterize the dynamical system with a number of parameters which is lower than the half of the dimensions used to reconstruct it with the highest accuracy.

State space reconstruction method outperforms the time-domain gait method used: best accuracies were 92% and 82%, respectively. Although the approach presented in this paper obtains a performance much higher than common gait identification methods, results should be taken as a preliminary study since database is composed of only 5 users and are not representative enough to be generalized. A more extensive database should be tested in order to confirm the abilities of the method. However, results show that relevant characteristics of gait can be obtained from the reconstructed space.

It should be noted that the algorithm is computationally expensive compared to the time-domain method. A real-time implementation of the algorithm would require a significant effort since large amounts of memory and computational capacity are used.

5. Conclusions

A methodology to identify people by gait has been tested and compared against a standard gait recognition method. It considers human gait as a dynamical system, whose attractor is reconstructed and characterized in order to recognize it. The characterization is performed by a spectral analysis of the reconstruction space, and it is tested into a small database of 5 people achieving an overall accuracy of 92.3%. A triaxial accelerometer located in the waist is needed to perform it.

Attractor's reconstruction is based on Taken's theorem. The different parameters involved in the reconstruction and the characterization are tested in order to evaluate the effect in gait identification. It is concluded from results that unfolding a whole orbit seems to provide the best identification, though in some cases unfolding a part of the orbit may be enough. Suitable window sizes for identification are those equal or larger than the orbit duration.

Further research is needed in order to validate in a more extensive database. Recently, a database of 50 people has been obtained and the abilities of the methodology are going to be validated. Future work will also try to use the same methodology for rehabilitation applications in order to detect, for example, the gait progress in a patient after a clinical intervention by measuring whether the attractor changes.

Acknowledgments
This work is supported by the Spanish project SENSORIAL (TIN2010-20966-C02-02) Spanish Ministry of Education and Science.

References

[1] K.M. Culhane, M. O'Connor, D. Lyons, G.M. Lyons. Accelerometers in rehabilitation medicine for older adults, *Age and Ageing* **34**-6 (2005), 556-560.

[2] A.P. Yazdanpanah, F. Faez, R. Amirfattahi, Multimodal biometric system using face, ear and gait biometrics, *Information Sciences Signal Processing and their Applications* (ISSPA), 2010 10th International Conference on.

[3] D. Gafurov, K. Helkala, and T. Søndrol, Biometric Gait Authentication Using Accelerometer Sensor, *Journal of Computers*, **1**-7 (2006), 51-59

[4] S. Sprager, D. Zazula, Gait Identification Using Cumulants of Accelerometer Data, *Proceedings of the 2nd WSEAS International Conference on Sensors, and Signals and Visualization, Imaging and Simulation and Materials Science* (2009)
[5] D. Gafurov, E. Snekkenes and P. Bours, Gait Authentication and Identification Using Wearable Accelerometer Sensor, *IEEE Workshop on Automatic Identification Advanced Technologies*, (2007)
[6] J. Mäntyjärvi, M. Lindholm, E. Vildjiounaite, S. Mäkelä, H. Ailisto, Identifying users of portable devices from gait pattern with accelerometers, *IEEE International Conference on Acoustics, Speech, and Signal Processing* (2005)
[7] A. Samà, F. J Ruiz , C. Perez, A. Catala, Gait identification by using Spectrum Analysis on state space reconstruction, *International Work-Conference on Artificial Neural Networks*, (2011)
[8] A. Samà, D. Pardo, J. Cabestany, A. Rodríguez-Molinero. Time Series Analysis of inertial-body signals for the extraction of dynamic properties from human gait. *Proceedings of IJCANN Conference*, (2010)
[9] L. A. Schwarz, D. Mateus and N. Navab, Multiple-Activity Human Body Tracking in Unconstrained Environments, *Lecture Notes in Computer Science*, **6169** (2010), 192-202
[10] J. Frank, Learning state space models from time series data, *Multidisciplinary Symposium on Reinforcement Learning,* 2009, Quebec
[11] T. Sauer, J. A. Yorke, M. Casdagli. Embedology. *Journal of Statistical Physics*, **65**-3/4 (1991) 579-616
[12] F. Takens, Detecting strange attractors in turbulence, *Dynamical systems and turbulence* **898** (1981) 366-381.
[13] H. Kantz, T. Schreiber. Nonlinear Time Series Analysis. Cambridge University Press; 2 edition (January 26, 2004)
[14] A. M. Fraser and H. L. Swinney, Independent coordinates for strange attractors from mutual information, Phys. Rev. A 33, 1134 (1986)
[15] M. B. Kennel, R. Brown, and H. D. I. Abarbanel, Determining embedding dimension for phase-space reconstruction using a geometrical construction, Phys. Rev. A 45, 3403 (1992).
[16] D. R Fredkin, and J. A. Rice. (1995), Method of false nearest neighbors: A cautionary note, Physical Review E, 51(4), 2950–2954.
[17] R. Vautard, Pascal Yiou, Michael Ghil, Singular Spectrum Analysis: A toolkit for short, noisy chaotic signals. Physica D 58 (1992), 95-126
[18] E. K. Antonsson and R. W. Mann, The frequency content of gait, *Journal of Biomechanics* **18**-1 (1985), 39-47.

Indexes

Keyword Index

Author Index