CONSTRAINT PROGRAMMING AND MULTI-AGENT SYSTEM MIXING APPROACH FOR AGRICULTURAL DECISION SUPPORT SYSTEM

Sami AL-MAQTARI *Habib ABDULRAB* *Ali NOSARY*
sami.almaqtari@insa-rouen.fr abdulrab@insa-rouen.fr ali.nosary@univ-rouen.fr
PSI laboratory – INSA of ROUEN
Place Emile Blondel - BP 08
76131 Mont-Saint-Aignan – France
URL: http://www.insa-rouen.fr/psi/
Tel: 0033 (0)2 35 52 84 70 - Fax: 0033 (0)2 35 52 84 41

ABSTRACT

The objective of this paper is to present the grand schemes of a model to be used in an agricultural Decision support System. We start by explaining and justifying the need for a hybrid system that uses both Multi-Agent System and Constraint Programming paradigms. Then we show our approach for Constraint Programming and Multi-Agent System mixing based on *controller agent* concept. Also, we present some concrete constraints and agents to be used in an application based on our proposed approach for modeling the problem of water use for agricultural purposes.
Keywords: Multi-Agent System (MAS), Constraint Programming (CP), Decision Support System (DSS), controller agent, water management.

1. Introduction

Water is the most vital resource in human life and a critical economic factor in the developing countries. And Yemen is considered as one of the most water-scarce countries in the world. According to worldwide norms, domestic uses and food self-sufficiency require up to 1100 m^3/capita/year. However, in Yemen the available water quantity amounts to little more than 130 m^3/capita/year [12]. Moreover, resources are unevenly distributed, 90% of the population has access to less than 90 m^3/capita/year. Table 1 shows a comparison of annual quota per capita between some countries in the region and the global average.

The decrement in annual water quota per capita in Yemen is due to (among other causes) the high population growth rate which has been about 3.7% (1980-1997 average) and is expected to be about 2.6% (1997-2015) [12].

Most water use goes for irrigation purposes [10]. The average share of agriculture in total water use is about 92% because of the rapid progress of irrigated agriculture in Yemen at a pace faster than any comparator country (see Table 2).

Table 1 Renewable Water Resources Per Capita

Country name	Renewable Water Resources Per Capita (m^3/capita/year)		
	1980	1997	2015
Egypt	1,424	966	735
Jordan	396	198	128
Morocco	1,531	1,088	830
Saudi Arabia	257	120	69
Yemen	246	130	82
World	10,951	8,336	6,831

Sa'dah basin is one of the four principle basins in Yemen and one of the most touched regions by water crises. Well inventory [11] shows a total of 4589 water-points in Sa'dah Region, of which 99.78% wells and the rest represents springs and other water-point (tanks, dams). These water-points are either owned by one owner (46.61%) or shared between two or more owners (53.39%). Agricultural wells represent over 90% of the annual water abstraction of the Basin.

Table 2 Water use share

Country name	Agriculture %	Industry %	Domestic %
Egypt	86	8	6
Jordan	75	3	22
Morocco	92	3	5
Saudi Arabia	90	1	9
Yemen	92	1	7
World	69	22	9

Well inventory shows also that for 82% of the wells is used for irrigation, while 1% is used for domestic needs and 0.5% for feeding water supply networks. About 16.5% of the inventoried wells are not in use. In which consider irrigation status we can find that only 2.64% of the sampled farmland is rainfed land while the rest (97.36%) is irrigated land.

The complexity of such situation requires reflect the need for a good decision support system. Such system has to take into account all necessary constraints such as the respect of shared water-points using agreement. It has also to model and to simulate the interaction between the different actors in the whole process such as the negotiations between consumers and water suppliers, and to model decision taking process, like the criteria and strategy of water allocation that are used by water suppliers. By making an analogy, constraint programming, therefore, looks a good approach in order to help finding a solution that satisfies the constraints of the problem, while multi-agent system approach can help in describing the interaction between the various actors in the model.

In the next sections we give a short introduction to the constraint programming and multi-agent system, after that we describe our approach for mixing both paradigms in order to model the problem of water using for irrigation purposes.

2. Constraint Programming

2.1. Introduction

Constraint programming is an emergent software technology for declarative description and effective solving of large, particularly combinatorial, problems. It is a programming paradigm in which a set of constraints that a solution must meet are specified rather than set of steps to obtain such a solution. A constraint is simply a logical relation among several unknowns (or variables), each taking a value in a given domain. A constraint thus restricts the possible values that variables can take; it represents some partial information about the variables of interest. The idea of constraint programming is to solve problems by stating constraints (conditions, properties) which must be satisfied by the solution.

2.2. Constraint Satisfaction Problem (CSP)

A Constraint Satisfaction Problem (CSP) consists of:

1. a set of n variables X = { $x_1, x_2, \ldots, x_n$ },
2. for each variable x_i, a finite set D_i of possible values (its domain),
3. and a set of constraints restricting the values that a set of variables can take simultaneously.

A solution to a CSP is an assignment of a value to every variable from its domain, in such a way that every constraint is satisfied. We may want to find: (i) just one solution, with no preference as to which one, (ii) all solutions, or (iii) an optimal, or at least a good solution, given some objective function defined in terms of some or all of the variables.

Thus, the CSP is a combinatorial problem which can be solved by search. Clearly, with a large number of variable simple algorithms of searching all possible combinations take a long time to run. So the researches in the constraint satisfaction area concentrate on finding ad-hoc algorithms which solve the problem more efficiently, especially by using techniques like global constraints.

2.3. Global constraint

A global constraint encapsulates several simple constraints [2], [3] and by exploiting semantic information about this set of constraints it can achieve stronger pruning of domains. Filtering algorithms for global constraints are based on methods of graph theory, discrete mathematics, or operation research so they make the bridge between these mathematical areas and search-based constraint programming with origins in artificial intelligence. [2] has proposed a dynamic view of global constraints. Such a dynamic global constraint allows adding a new variable(s) during the course of problem solving and removing this variable(s) upon backtracking. Thus, a dynamic global constraint can be posted before all the constrained variables are known which brings the advantage of earlier domain pruning mainly for a system where not all information is necessarily known a priori.

2.4. Over-Constrained Problems and Constraint Hierarchies

In many cases, a solution of CSP does not exist, and we can not make a valuation of variables that satisfies all the constraints. Constraint hierarchies [9] were introduced for describing such over-constrained systems by specifying constraints with hierarchical strengths or preferences. It allows us to specify declaratively not only the constraints that are required to hold, but also weaker constraints at an arbitrary but finite number of strengths. Weakening the strength of constraints helps to find a solution of previously over-constrained system of constraints. Intuitively, the hierarchy does not permit to the weakest constraints to influence the result. Constraint hierarchies define the so called comparators aimed to select solutions (the best assignment of values to particular variables) via minimizing errors of violated constraints.

2.5. Constraint Programming and agricultural water management

Jaziri [7] has proposed a methodology for constraints and optimization modeling. We can note that the constraints in the classical definition of CSP are relations between sample variables. However, in a complex system like agricultural water uses, the constraints represent relations between different instances of the system (parcels, resources …) acting on the variables that characterize these instances (parcel crop, demanded water quality …).

According to [1] we can distinguish three layers: constraints, instances and variables. These layers are divided according two different points of view: user level and system level. At user level, a constraint represents a condition that links some system instances. At system level, the constraint is defined as a restriction over the values of a set of simple variables that characterize the instances linked by this constraint.

In the case of agricultural water use, the user expresses the system constraints such as water provision which is a relation between the supplier and the consumer. This relation is translated at system level as constraints relating various variables such as consumer required water, supplier available water quantity, transport mean capacity, etc…

3. Multi-Agent System

3.1. Introduction

The Agent-Oriented (AO) approach gives the ability to construct flexible systems with complex and sophisticated behavior by combining highly modular components. These components represent agents having autonomy and interaction characteristics.

What is an agent? The term agent has many definitions. According to Wooldridge [13] an agent is a software system that is (i) situated in some environment, (ii) capable of autonomous actions in order to meet its objectives and (iii) capable of communicating with other agents. From this definition we can say that an agent is an entity that can act and react in his environment and interact with other agents.

A multi-agent system is made up of a set of several agents (representing different tasks and/or entities in the system) that exist at the same time, share common resources and communicate with each other. For simplicity a multi-agent system can be viewed as a network of agents (problem solvers) coupled lightly, who work together to solve problems that are beyond their individual capacities [5].

The research on the agents is also a research on: (i) Decision - what are the mechanisms of the agent decision? What is relation between their perception, their representations and their actions? How the agents break down their goals and tasks? (ii) Control - what are the relations between agents? How are they coordinated? This coordination can be represented as a cooperation to fulfill a common task or as a negotiation between agents having different interests. (iii) Communication - what types of message do they exchange? What syntax these messages obey?

3.2. MAS and the simulation of resource management

The simulation of the management of common resources poses the problem of correlation between groups of agents and dynamic resources. In the multi-agent system paradigm we look at the simulated system from a distributed and cooperative point of view.

In the domain of water use for agricultural purposes, we can find various actors (different farmers, resource managers …). Using multi-agent system paradigm allows us to simulate these actors' decision mechanisms and their evolution, the interactions, the negotiations, and the cooperation between these actors in the model.

4. MAS & CP mixing approach

4.1. Introduction

We can notice that the model of the simulated system using Constraints Programming is built as a set of variables that represent the simulated system variables, and different constraints relating between these variables. All these constraints will be inserted into a solver to be manipulated and treated together as a whole unit in order to find and assign values to the system variables. In the other side, in the multi-agent system, agents are mainly characterised by the autonomy, i.e. each agent tries independently to achieve its own goal. The agent could interact, cooperate and/or negotiate with other agents either directly or via its effects on the environment. Combining the both paradigm defines the Distributed CSP.

4.2. Distributed CSP

Distributed constraint satisfaction problems (DCSP) are an appropriate abstraction for multi-agent cooperative problem solving [6]. They are characterized by multiple reasoning agents making local independent decisions about a global common constraint satisfaction problem (CSP).

In a DCSP, the variables are distributed among agents (see Figure 1). Yokoo [14], [15] has proposed solving distributed CSP by using an asynchronous backtracking algorithm. This can be done by allowing agents to run concurrently and asynchronously. Each agent give values for its own variables and communicates these values with relevant agents.

We can note from Figure 1 that the constraints are represented by oriented arcs between the agents. Agents propose their variables values according to the oriented arcs. In this example A_3 receives values propositions from both agent A_1 and A_2 and test them according their respective constraints with its own variable possible value. Its response is a result

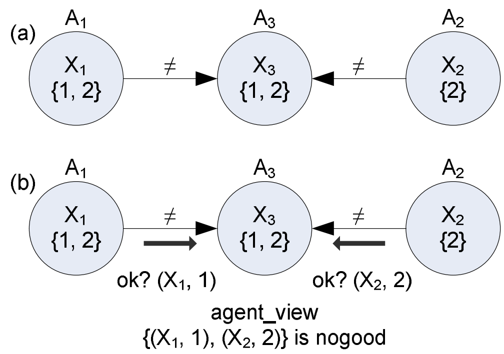

Figure 1 an example of DCSP

4.3. Another approach of MAS & CSP mixing

We propose here another approach for DCSP resolution. It consists of adapting the methodology like that proposed by [7] for constraints modelling with multi-agent system paradigm. Our approach is based on a special kind of agents called *controller agent* that takes in charge the responsibility of verifying the constraint satisfaction. We will show this approach through the following sections:

4.3.1. Our model

Our approach [1] of mixing MAS and CSP uses the concept of a controller agent to verify the constraint satisfaction. If we compare between Figure 1 and Figure 2 we can note that the agents in the system (A_1, A_2, and A_3) do not communicate directly the proposed values of their variables. Instead, they send these values to controller agents (C_1 and C_2) who verify these values according to the constraint which they hold.

Controller agents report variable agents of the constraints satisfaction state and inform them of the accepted or refused values. If we take the example shown in Figure 2, at first (a) the variable agents send their value proposition to the controller agents. Then (b) each controller agent verifies the validity of its own constraint according these propositions. C_2 accepts the values sent by A_2 and A_3; therefore it sends back an acceptance message. While the values sent by A_1 and A_3 do not satisfy the controller C_1, therefore it sends a message of refuse back to them and wait for other propositions. In (c) A_3 has already satisfied a constraint and prefers to wait a little before proposing another value that may affect both constraints, while A_1 can freely propose another value from its variable domain. This new value is verified by C_1 (d) which sends back an acceptance message.

Using a controller agent for verifying a constraint satisfaction allows separating constraints verifying process from the functionality of other agents.

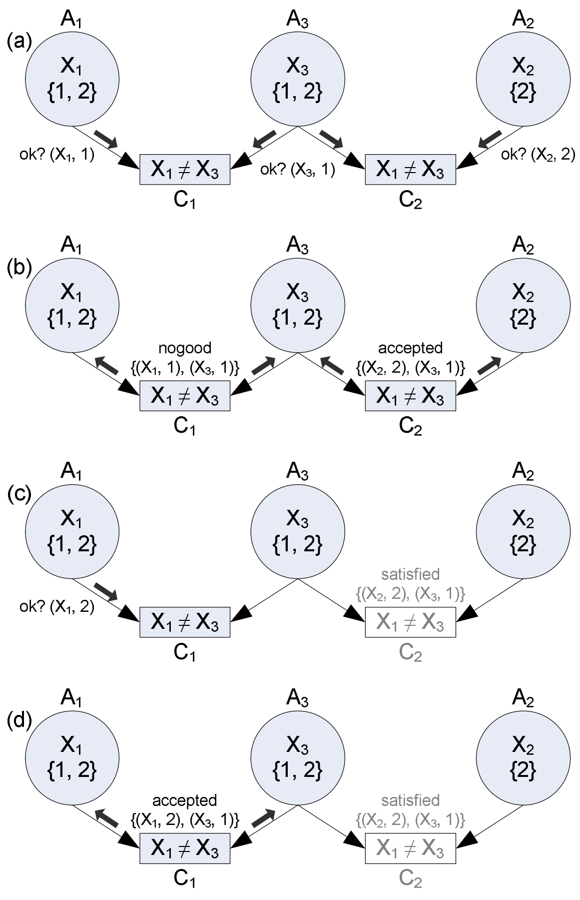

Figure 2 an example of our mixing approach

4.3.2. Approach application for agricultural water use management

Agricultural water use management implies various actors constrained by different constraints. The constraints in such domain are represented by relations between the different actors (or actors' attributes). For example, shared water-points user should have an agreement for water use (they are constrained by this agreement). Applying our approach for agricultural water use is described as follows:

4.3.2.1 Approach application for agricultural water use management

In Sad'ah basin we can find:

- Different modes of water consuming: agricultural (crop et livestock), domestic, industrial, urban et municipal …
- Different types of water resources (water-points), some of them are owned by farm owner and others are shared between several farmers.
- Several means of water transport.
- Several irrigation techniques.

We can note here that there are a lot of parameters need to be manipulated in order to achieve a stable and durable situation of water consuming. To simplify and minimize the problem parameters we can note that the agricultural water use represents about 92% of the total water use in Sad'ah basin, so we can consider (for the moment) the use of water for agricultural purposes only. Another note is coming from the fact that wells represent 99.7% of water-points used for irrigation purposes. So we can neglect other water-points. This can help to reduce first and second points of the above description into: Two ownership types of wells used for irrigation purposes. Figure 3 shows a general UML diagram of the proposed model.

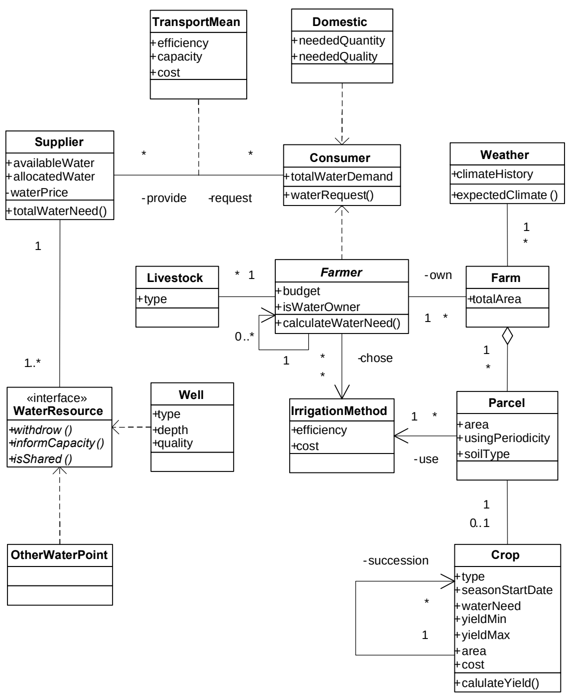

Figure 3 General UML diagram of the proposed model

Water authorities want to keep water use at its durable limits. Farmers who own their water resources want firstly to increase their profits and try not to overexploit their own water resources. While farmers who share common water-points want to have more convenient access to water with acceptable costs and keep their profitability. Other farmers need only to be satisfied by the proposed solution. Water and local authorities encourage the use of more efficient water transport means and irrigation techniques. Framers are willing to use water saving technique but they do not want to affect their profitability with extra costs.

We can cite here the needs of the different deciders in the system:

4. Increasing the profitability (water owner farmers).
5. Limiting water overexploiting (water owner farmers).
6. More convenient water access (shared-water farmers).
7. Keeping the profitability (shared-water farmers).
8. Framers satisfaction (farmers).
9. Achieving durable water use (water authorities).
10. encouraging the use of water saving techniques (water and local authorities, …)

These needs can be translated into the following objectives:

1. Enhancing water exploiting (needs 1, 3, 6, and 7).
2. Keeping the profitability (needs 1 and 4).
3. Framers satisfaction (need 5).

The aspects of these objectives can be shown as follows:

1. Enhancing water exploiting: can be view from two different aspects.
   - Reducing water wastage at irrigation technique level (aspect 1).
   - Reducing water wastage at the transport means level (aspect 2).
   - Enhancing water sharing agreement (aspect 3)
2. Keeping the profitability: one aspect (aspect 4)
3. Framers satisfaction: satisfaction in profitability (same aspect 4).

These aspects have the following criterions:

1. Reducing water wastage at irrigation technique level: the waste water can be estimated by considering the water used for irrigation and the actual crop water requirements. It is also related to the efficiency of irrigation techniques.
2. Reducing water wastage at the transport means level: it is a function of the efficiency of transport means.
3. Enhancing water sharing agreement: it can be calculated by considering farmer budget and shared water cost.
4. Keeping the profitability: it is a function of crop yield and planted area in each farm and the cost of irrigation and transport means.

We can deduce finally the attributes of the system for evaluating these criterions:

1. Crop water needs:
   crop.waterNeed (criterion 1).
2. Parcel water use:
   totalWaterUse (criterion 1).
3. Irrigation method efficiency:
   irrigationMethod.efficiency (criterions 1 and 4).
4. Irrigation method cost:
   irrigationMethod.cost (criterion 4).
5. Transport means efficiency:
   transportMean.efficiency (criterions 2 and 4).
6. Transport means cost:
   transportMean.cost (criterion 4).
7. Farmer budget:
   farmer.budget (criterion 3).
8. Shared water price:
   supplier.waterPrice (criterion 3).
9. Crop yield:
   crop.yieldMax, crop.yieldMin (criterion 4).
10. Parcel planted area:
    parcel.area (criterion 4).

4.3.2.2 The constraints in the application

As we mentioned in section 2.5, the constraints in our case represent relations between different instances of the system. We can cite here some constraints in the model:

1. Crop water requirement: every crop has its own water requirement that must be satisfied. This requirement varies according to the start date and the current stage of crop evolution.
2. Compatible soil type: the soil type of a parcel determines what kind of crop can be planted on this parcel.
3. Parcel crop type preference: some farmer can accept only planting specific types of crop, but he may also accept a reasonable proposition out of these choices (constraint relaxation). This constraint can be seen as specifying a specific domain for each parcel.
4. Profitability of crop type: farmers privilege the crops with the highest profitability. This implies the water prices and crop planting cost (land preparation, seed cost, labor cost …).
5. Crop succession: some type of crops can not be planted in the same land without knowing the previous planted crop type in the same land.
6. Water sharing agreement: shared water points can not be used except in a certain quantity and in a certain slice of time.
7. Transport means capacity: even if we have sufficient quantity of water we can not transport a quantity superior to transport mean capacity.

4.3.2.3 The agents in the application

The objective of the model is to assign a crop to every parcel in a way that respects the constraints of the system and try to optimize water exploitation. We can distinguish here three main types of agents:

1. Farmer agent: this agent takes in charge the following tasks:
   - Choosing its parcels' crops type and calculating their water requirements.
   - Choosing the irrigation method.
   - Negotiating with other farmer agents (who share the same water-points).
   - Demanding water supplying from supplier agents.

2. Supplier agent: it represents the water resources manager. It decides water quantity and quality to be provided to each consumer (farmer in this case).
3. Controller agent: they are the agents who control the validation of constraints of the system. In the following section we will explain controller agent functionality in more details.

4.3.3. Controller agent in our approach

Controller agent represents essential party of our approach in mixing Multi-Agent System and Constraints programming paradigms. We have seen in Figure 3 a general UML diagram of proposed model. In our approach, each controller agent is attached to a constraint in the system. The controller agent has the responsibility to assure the satisfaction its constraint according to the constraint strength level (see section 2.4). It has the ability to decide if either (i) the constraint is satisfied, (ii) the constraint is not satisfied, or (iii) the constraint is not totally satisfied but it is accepted (for the constraints with a low level hierarchy or week strength). This gives the model a kind of flexibility because the system can achieve a stable state and have evaluation values for its variable even if some of its (week) constraints are not satisfied.

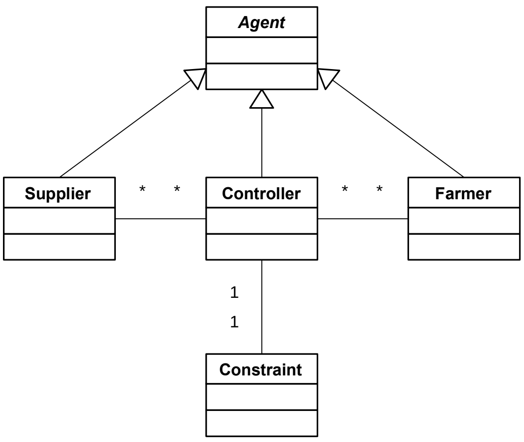

Figure 4 model agents and their inter-connection

As shown in Figure 4, the system agents are inter-connected between them. They are associated by constraint that rules the relation between them. Farmer and supplier agents send their variables values to a controller agent according to their respective constraints. Each controller agent communicates then its constraint satisfaction state with both consumer and/or supplier agents.

Note that a constraint does not link only farmers and suppliers; it may link between different suppliers or different farmers only. In fact it joins only the agents who are related to (or constrained by) the controller constraint. For instance, crop succession is represented by a controller who is linked to the only one farmer but it constrains the farmer decision on what crop to be chosen for the next period of plantation.

Assigning a controller agent for a constraint makes it easy to manipulate system constraints. When a constraint is defined we actually define not only the variables and the constraint relation between them, but also the agents which are involved in this constraint. In other word, the agents which are intended to participate in a constraint have to provide the variable required by this constraint.

If we take the transport means capacity for example we can define this constraint as follows: the sum of water allocated by the water supplier (k) to all consumers connected by the same transport mean should not exceed this last one capacity. In this case, controller agent that checks this constraint would have a water supplier and some consumer (farmers in our case) as participants and every time a supplier try to provide his consumers with a quantity greater than transport mean capacity the controller send a message to it refusing this value.

The modelled constraints take many variables and have a sort of symmetry. This symmetry allows us to investigate the use of global constraint in order to make the resolution process more efficient. For example, although we have defined one constraint representing crop water requirement, this constraint will be instantiated for every simulated parcel. So the verification of this constraint satisfaction is in fact the verification of all instants of this constraint type. This implies the processing of very large number of constraints, and this is another aspect of the usefulness of global constraint techniques. Equally, as proposed by [2], [3], the use of dynamic global constraint allows adding new variables to the system dynamically. This looks very useful for a system where not all the information and variables values are entirely known.

5. Conclusion and perspectives

Mixing the capabilities of Constraint Programming and Multi-Agent System represents an interesting approach for constructing a decision support system for the management of water use for agricultural purposes. The advantage of this approach is the fact of dealing with CSP is not done as a group of constraints and variables that have to be processed as a whole together. The solution of the system is emerged by the interactions between the different actors. Such approach allows endowing controller agents with the ability of controlling system constraints and –as a result- satisfying the constraints locally. It facilitates adding not only new constraints in the system, but also other sort of consuming mode.

We have the intention to detail and extend the system components (the different constraints to be implemented, and the various agents to be added). Completing this model allows us not only to implements this model and to test it on the real situation of Sa'dah, but also to extend the model in order to be used with other types of consumer (domestic, industrial …) and their relative new constraints.

6. References

[1] S. Al-Maqtari, H. Abdulrab, and A. Nosary, "A Hybrid System for Water Resources Management", GIS third international conference & exhibition, Sept. 2004.

[2] Roman Barták, “Dynamic Global Constraints: A First View”, Proceedings of CP-AI-OR 2001 Workshop, pp.39-49, Wye College, April 2001.

[3] Roman Barták, “Dynamic Global Constraints in Backtracking Based Environments”, Annals of Operations Research, no.118, pp.101-119, 2003.

[4] Christian Bessiere, E. Hebrard, Brahim Hnich, and Toby Walsh, “The Complexity of Global Constraints”, American Association for Artificial Intelligence 2004.

[5] E. H. Durfee, T. A. Montgomery, “MICE: A Flexible Testbed for Intelligent Coordination Experiments”, Proceeding of 9th International AAAI Workshop on Distributed Artificial Intelligence, pp.25-40, 1991.

[6] William S. Havens, “NoGood Caching for MultiAgent Backtrack Search”, American Association for Artificial Intelligence (www.aaai.org) June 1997.

[7] Jaziri Wassim, “Modélisation et gestion des contraintes pour un problème d'optimisation sur-contraint - Application à l'aide à la décision pour la gestion du risque de ruissellement”, PhD Thesis, INSA-ROUEN - France, 2004.

[8] M. Le Bars, J.M. Attonaty, & S. Pinson, “An agent-Based simulation for water sharing between different users”, Xth Congress of the European Association of Agricultural Economists (EAAE), Zaragoza, Spain, 2002.

[9] Hana Rudova, “Constraint Satisfaction with Preferences”, PhD Thesis, Faculty of Informatics, Masaryk University, Brno, Czech Republic, 2001.

[10] TECHNIPLAN, Unigeo & Technocenter, “Studies For Regional Water Management Planning - Agricultural Survey in Sa'dah Water Region - Final Report”, Rome and Sana’a, July 2002 Re-edited June 2004.

[11] TECHNIPLAN, Unigeo & Technocenter, “Studies For Regional Water Management Planning - Well Inventory in Sa’dah Region - Final Report”, Rome and Sana’a, July 2002 Re-edited June 2004.

[12] Christopher Ward, Satoru Ueda and Alexander McPhail, “Water Resources Management in Yemen”, January 2000.

[13] M. Wooldridge, “Agent-based software engineering”, IEEE Proceedings on Software Engineering, vol.144, no.1, pp.26-37, February 1997.

[14] Makoto Yokoo, Toru Ishida, Edmund H. Durfee, Kazuhiro Kuwabara “Distributed constraint satisfaction for formalizing distributed problem solving”, 12th IEEE International Conference on Distributed Computing Systems, pp.614-621, June 1992.

[15] Makoto Yokoo, Edmund H. Durfee, Toru Ishida, & Kazuhiro Kuwabara, “The Distributed Constraint Satisfaction Problem: Formalization and Algorithms”, IEEE Transactions on Knowledge and DATA Engineering, vol.10, no.5, pp.673-685, September 1998.