Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Applying Dynamic Data Mining on Multi-Agent Systems

I. J. Benítez J.L. Díez P. Albertos

Departamento de Ingeniería de Sistemas y Automática
Universidad Politécnica de Valencia
Valencia, Spain. P.O.Box 22012 E-46071
(e-mail: igbesan@isa.upv.es, jldiez@isa.upv.es, pedro@isa.upv.es)

Abstract: A new perspective on analysis of large-scale Multi-Agent Dynamic Systems is presented. The aim is to capture the global trends at a glance, by the use of dynamic data mining techniques, which group agents according to similar characteristics or behavior, measuring and recording how the different trends evolve through time. This methodology is presented and an example with a simulated dynamic Multi-Agent System is included.

1. INTRODUCTION

Multi-Agent Systems (MAS) constitute an interesting approach to deal with complex systems, either in simulated environments (Bar-Yam [1997] and Martínez-Miranda, Aldea & Bañares-Alcántara [2003]) or to control and maintain industrial applications (Castro & Oliveira [2005] and Pereira & Carro [2007]). An agent can be considered, in a generic definition, as an autonomous entity with a limited knowledge of its surrounding or environment, which reacts according to its objectives (predefined or learned), the information received from the environment and the interactions with other agents (Ferber [1999] and Bonabeau [2002]).

A system of agents constitutes a collective whose characteristics (position in the case of moving agents, or other states or variables) can vary along the time. In a simulated environment, all the information from the agents is available at each cycle. However, different scenarios must be considered where the information flow among agents is asynchronous, only dependent of the events produced. The data about the states or characteristic variables of all the agents is gathered in a discontinuous, asynchronous way, including different delays as the information from each agent is sent through a different channel.

Therefore, a set of dynamic data is considered. The variables dataset is updated whenever new information from any agent is available. A global knowledge about the system at a given time will not coincide with the same scenario when new data is stored at a different time. The technique proposed in this paper is the Dynamic Data Mining (Crespo & Weber [2005]), being able to take advantage of new available data, combining previous and new knowledge (when available) about the system in an appropriate way.

This technique is suitable to measure, record, and evaluate the trends of groups of agents sharing similar characteristics, being part of a dynamic MAS. Application examples considered, appropriated for this methodology, would include networks of autonomous dynamic agents (Ferrari-Trecate, Buffa & Gati [2005]), such as mobile robots or vehicles. These systems constitute a decentralized network where each agent has to accomplish a specific objective, reacting in a determined way whenever another agent approaches it, or an obstacle is reached. Both communication for the first case, and collision avoidance protocols for the second, must be implemented and combined with the movement due to the accomplishment of the specific goal assigned to the agent. The behavior of these networks include different schemes, such as a'follow the leader' implementation, where each agent has to follow the agent in front of it, keeping a security distance, or flocking formations, imitating the collective behavior in natural systems, where all the agents have the same goal and a consensus is reached among them concerning heading angles and separation distances (Cruz, Mcclintock, Perteet, Orqueda, Cao & Fierro [2007]).

The objective is to analyze the collective motion of the MAS, to track and somehow infer from partial information the collective dynamics of the agents. In some cases, for simulation purposes, the behavior of each agent can be modeled, including the effect of the interaction with the other agents, therefore a model of the MAS can be obtained, by means of state-space equations. See, for instance, (Ferrari-Trecate et al. [2005]). However, in other cases the model is unknown, and the only available information are measurements from the state of the different agents at different times (such as, for instance, a signal sent by the robot indicating its position). The methodology of dynamic data mining allows the depiction of a global picture of the main trends of groups of agents with similar behaviors, which includes the evolution of the different groups, how they have formed, split from bigger groups, or joined to others. This methodology can be useful when no information about the goals or missions of the different agents is known, and it has to be inferred from the resulting trends. The same can be applied to other measured variables (not only position and not only from moving agents), such as the case of static agents in a network, with different tasks assigned, that periodically report their internal state, output values, or working points (Castro & Oliveira [2005] and Naso & Turchiano [2004]). Implementations of this kind vary from software agents in nodes of a distributed network (Nwana [1996]), to agent-based manufacturing systems or distributed real-time embedded systems (Pereira & Carro [2007]). The dynamic data mining methodology would allow to track and observe the evolution of the states of all the agents, giving a global view of similar behaviors or their convergence to a set of different states. Each time the information about the state of some of the agents is available, the information about the behavior patterns and the trends of the different groups of agents can be computed. Despite the fact that a complete information from all the agents at the same time, at each cycle, would be the desirable scenario, partial information can also be used to update the database and to perform a new analysis, which will result in a modification of the current trends or patterns detected at the MAS.

In the rest of this paper, a methodology to deal with this proposal of agents' trend evaluation and registering is presented. A simulation example where this methodology is applied on a simulated dynamic MAS is reported. A conclusions section is included, along with a list of used references.

2. DYNAMIC DATA MINING

The objective is to capture the global view of the different trends that may arise at a large-scale MAS through time, and how they evolve according to the different inputs, initial states and nature of the agents. The proposed methodology is based on the data mining processes. Different techniques can be applied when dealing with data mining, in general, and dynamic data mining in particular. Some of the most used approaches are: rule-based (for instance, fuzzy) systems (Chan & Au [1997]), neural networks (Zhang, Fraser, Gagliano & Kandel [2000]), decision trees (Lutu [2002]), and clustering methods (Han & Kamber [2001]).

Fuzzy models are widely used because of their universal function approximation capabilities and the parallelism to human reasoning processes (Wang, Yang & Muntz [1997] and Díez [2003]). Moreover, fuzzy rule-based systems, neural networks, and clustering techniques have a strong relationship, provided that rule extraction tasks can be given or provided by an expert but, in general, methods for solving the rule extraction problem (Nelles [1999]) are based on genetic algorithms, neural networks, templates, or clustering techniques. Rule extraction using clustering provides, among other advantages (Duda, Hart & Stork [2000]), systematic methodologies for fuzzy identification of models or classifiers (Babuska [1998], Díez & Navarro [1999], Emami, Türksen & Goldenberg [1998] and Sugeno & Yasukawa [1993]), different available techniques for dealing with quantitative and/or qualitative data (Andritsos [2002]), and the possibility to apply compatible cluster merging (Krishnapuram & Freg [1992]) or possibilistic partitions (Krishnapuram & Keller [1993] and Díez, Sala & Navarro [2006]) to a defined algorithm.

Previous works on the field of dynamic data clustering can be found for hierarchical (Karypis, Han & Kumar [1999]), or objective function (mainly fuzzy) approaches (Joentgen, Mikenina, Weber & Zimmermann [1999]). In Crespo & Weber [2005], the concept of *cycle* (period between the creation of a classifier and its update) is defined. The main idea behind this concept is that fuzzy clustering techniques are used to determine static attributes to dynamic data, but attributes would be updated (making use of new data when available) after a period of time.

The use of fuzzy logic (Zadeh [1965]) is recommended throughout this paper, not only as the main concept behind the algorithms in both stages (object assignment and clustering), but also when some variables need to be graded, in order to establish the criteria for new prototypes to be created or existing ones to be removed. The possibilistic factor (Díez et al. [2006]) is also mentioned, as a more versatile option that allows a more accurate assignment and membership recognition of an object to one or various prototypes.

3. PROCEDURE FOR THE DYNAMIC MINING ON MULTI-AGENT SYSTEMS

The process that is going to be presented has been designed as a process of continuous evaluation, being split in two stages, each time that new measures are available and the analysis is relevant. In a first step an object assignment is produced, of the available data from each agent to the existing prototype or prototypes that it fits best. A reconfiguration of prototypes is carried out from this assignment. Next, at the second step, the new configuration will be redefined and fine tuned by the use of clustering techniques.

3.1 Description of the methodology

Step 1: Membership assignment At this stage, whenever new data measures are available, a process starts where the data from each agent are compared with the existing base of prototypes (the ones that were produced in the previous cycle), to check the membership of the agents to the current set of prototypes, by a Pattern Recognition System (PRS) (Maravall [1993]). Usually the Euclidean distance is used as a similarity measurement, though this is not the unique (nor most recommended) option.

This assignment results in a membership grade of each agent to one or more prototypes to more or less extent, or no membership to any existing prototype at all. Three different options can be considered:

- Exclusive membership. Each object falls within the influence area of an existing prototype, therefore the object is said to belong to that prototype and does not belong to any other.
- Non-exclusive membership. Fuzzy logic techniques allow an assignment of different membership values of each data object to all the existing prototypes. The more the object is similar to a prototype, the higher its membership grade will be to that prototype. The sum of all the membership values of a same data object to all the existing prototypes must equal 1.
- Non-exclusive possibilistic membership. The fuzzy possibilistic option allows that the different memebership values of a data object to all the prototypes sum different than 1, which means that each data object can have a value of membership to a prototype according to its similarity, not restricted to an overall adequacy to all the prototypes. This one seems to be the most versatile option.

As a result of this process, an assignment is done of the measured data to a set of existing prototypes, as depicted in the diagram in Fig. 1. The data can be found to match into one of the current prototypes to a more or less extent, but usually, when dealing with a dynamic system, the values of the objects may have changed since the last measure, therefore the prototypes will not cover the new domain of all the new objects data. At this point, a procedure for prototypes updating can be started, following a computation in two steps:

- Data objects not matching any of the existing prototypes can be found. In this case, a new prototype is created, by defining one of the data objects as the centre or represen-

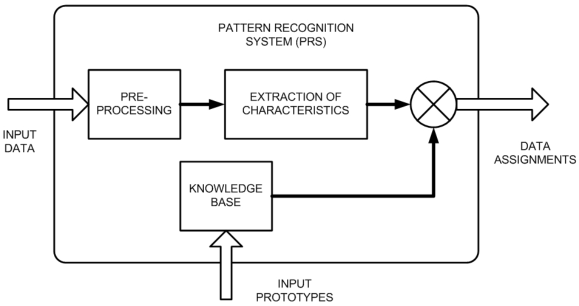


Fig. 1. Pattern Recognition System (PRS)

tative element of all the group. A new prototype is then created and added to the list.

- Prototypes can be found that have lost all the data belonging to them. These prototypes are thus "empty" and should be removed. Their characteristics are stored (in case of future re-apparitions) and the prototypes are removed from the list due to obsolescence.

The result will be a set of prototypes (defined by their centres) where the obsolete ones are excluded from the analysis, and new ones are defined according to necessity. A second stage of clustering will refine this recognition of prototypes.

Step 2: Clustering A fuzzy clustering algorithm, such as the FMLE, GK, or the FCM (Bezdek [1981]), is suggested to be applied on the data contained in the database from all the agents at all the instants of time. These algorithms group objects according to a similarity measure (usually the Euclidean distance, see Jain & Dubes [1988]), giving thus a powerful tool to capture the dynamic of the whole system at a glance. The parameters needed for the algorithms to run usually are the number of expected or desired clusters and an initial configuration of centres or membership grades. When no information about the final classes is available, the initial configuration is chosen based on a random computation. In this case, the set of prototypes obtained from previous cycles at the pattern recognition stage is used to compute the initialization state of the clustering algorithm.

As the algorithm converges from this initial configuration, the final result will refine the centres of the prototypes and the membership values of each agent to the detected prototypes, differing from the initial configuration in an extent which will be related to the accuracy of the previous estimation of prototypes. The resulting patterns inform about the variation in the characteristics of a dynamic MAS as they were from the last cycle where the analysis was done. Therefore, not only information about the different trends is recorded, but also about the *variability* of the MAS through time.

When dealing with qualitative data, clustering algorithms based on quantitative distances should not be used, and other solutions be applied instead (Andritsos [2004]). Clustering algorithms have been described in the literature that are designed to analyze qualitative data, based on other similarity measures different than the Euclidean distance, such as the interconnectivity and closeness at the CHAMELEON algorithm (Karypis et al. [1999]), the overlap metric at the fuzzy K-modes (Ng & Wong [2002]), or the entropy at the COOLCAT (Barbará, Li & Couto [2002]). Also algorithms specifically dealing with spatial clustering are available (Wang et al. [1997]).

With the result of the clustering algorithm, A further refinement of clusters and membership can be done at this point, allowing some possibilities, such as forgetting factors for the weight of the detected clusters from previous measurements, or a normalization procedure on the resulting clusters:

- Clusters with large areas can be split in two.
- Clusters whose objects share similar characteristics can be merged.

The new centers for created prototypes can be chosen and left ready for the next cycle. The next step would be to capture the characteristics of each prototype. Different variables can be considered like, for instance, a *health* variable. Given a group of N agents, from where information is measured and evaluated at each cycle or iteration $k = \{1 \dots n\}$, and a number of c detected prototypes is revised, the *health* variable is defined as a value related to the number of agents belonging to the same pattern or prototype. The value of this variable can be obtained from a fuzzy logic evaluation based on membership functions. Each detected pattern behavior has its own health variable, and an update of the values would be produced with each new cycle, following the formula given in (1), where Δs is a function that computes the variation in data membership to prototype $d = \{1 \dots c\}$ at each iteration.

$$s_{d_{k+1}} = s_{d_k} + \Delta s \tag{1}$$

Other variables to be considered are:

- The *area* of influence of a prototype.
- Its *density*, or number of objects per volume unit.
- *Statistical* information from the objects belonging to that prototype, such as the mean and the standard deviation.

The prototypes could be considered as a new MAS with "condensed" information enclosed (space and time). Registering allows to keep a track and observe how the different trends evolve, giving the possibility to build a polynomial model to estimate the behavior of the group of patterns, and therefore to be used as a prediction model to evaluate the trend of the overall system (see Fig. 2).

3.2 Resume and pseudocode

Concerning the process of dynamic information discovery, there is a need to re-evaluate the global knowledge about the system each time the information about the agents is available. A final step is, then, to consider how the relations could be established among the different resulting scenarios in different instants of time, in order to discover and record trends in the data objects. Figure 2 summarizes the whole process of the previously described dynamic mining. To help understanding the steps in the analysis, the pseudocode for the implementation of this methodology is reported:

```
Do for n times
  Update database of agents adding all
   the new available data
  Perform a Pattern Recognition System
   on the current base of patterns
   or prototypes
  Create or remove new prototypes
   according to the membership of the
   updated database
  Apply a clustering algorithm on the
   updated database, setting the
```

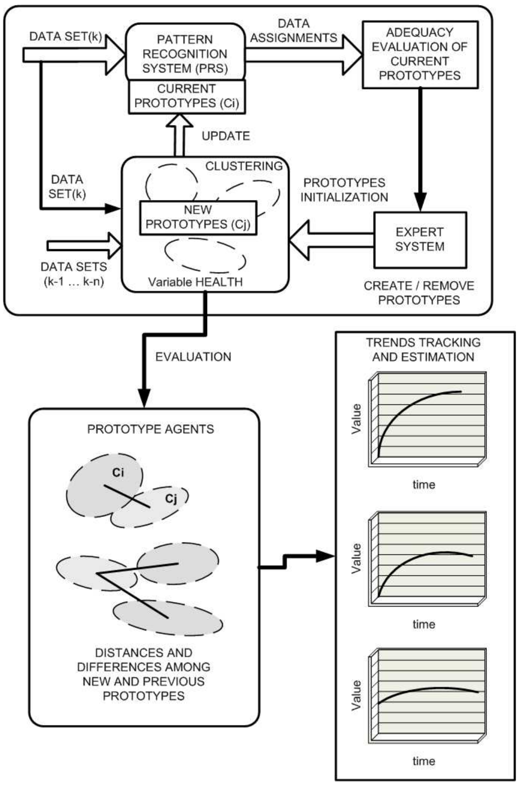


Fig. 2. Clustering process, tuning of prototypes and trends computation

```
   updated base of prototypes as the
   initial clusters to be refined
  Merge or split clusters according to
   their size, density or other variables
```

4. EXAMPLE

The present methodology has been designed for a full range of different, large-scale environments, such as industrial distributed systems where the different nodes can be considered as agents with local processes attached. However, to serve as an experiment study, a specific MAS has been designed. Based on a previous work (Díez, Benítez & Albertos [2005]), a model of dynamic agents is simulated at an unbounded environment.

4.1 Design of a Dynamic MAS

A MAS has been designed with a pure reactive behavior, in resemblance to cellular automata (Bar-Yam [1997]). Agents are thought to be autonomous entities that behave through the actuators or outputs, according to information gathered through sensors from interaction with the environment and the other agents (Russell & Norvig [2003]). In this case, the agents are modeled as units with a local influence area, delimited by a circle of radius r, and three characteristic variables: their position (x and y values), and a parameter m of attraction or repulsion for other agents. The objective is to simulate a dynamic environment where, depending on the initial configuration or spatial distribution, the agents move searching either to be close one to each other, either to be as far as possible from the rest of agents, according to the value of the parameter m. The equation that defines the behavior of each agent i can be expressed by (2).

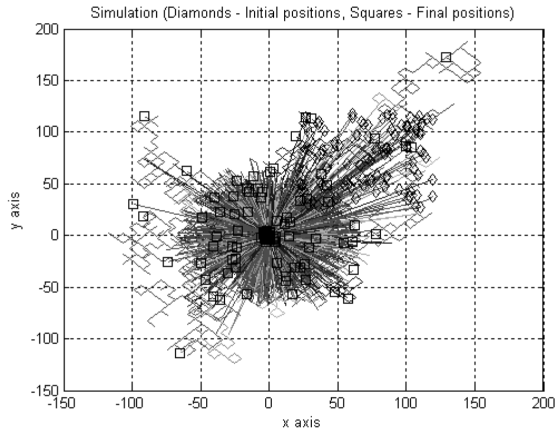


Fig. 3. Simulation example of the designed MAS

Only the agents that fall within the influence area of an agent at each cycle have an effect on it, which translates in a movement of the agent, towards or against the other agents, depending on the parameter m.

$$\begin{cases} x_i(k+1) = m_i \dfrac{1}{2} \displaystyle\sum_{j=1}^{n} (x_j(k) - x_i(k)) \\ y_i(k+1) = m_i \dfrac{1}{2} \displaystyle\sum_{j=1}^{n} (y_j(k) - y_i(k)) \end{cases} \tag{2}$$

The condition for any agent j to be inside the influence area of an agent i is defined by the equation of a circumference, such as (3).

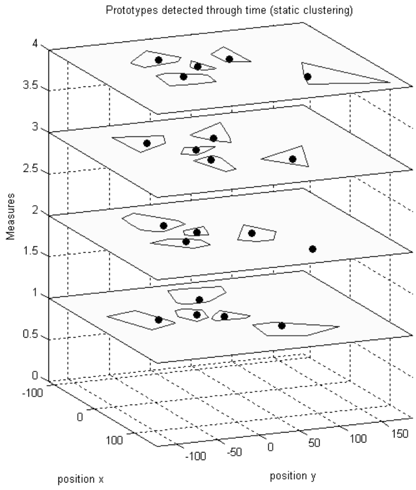
$$(x_j - x_i)^2 + (y_j - y_i)^2 \leq r^2 \tag{3}$$

In order to avoid a rapid decay of the dynamics, the agents that search other agents and do not find any other inside their influence area at a given cycle k, are assigned a movement towards a random position inside their influence area. Figure 3 depicts a simulation of 100 agents for 100 cycles of this kind of MAS.

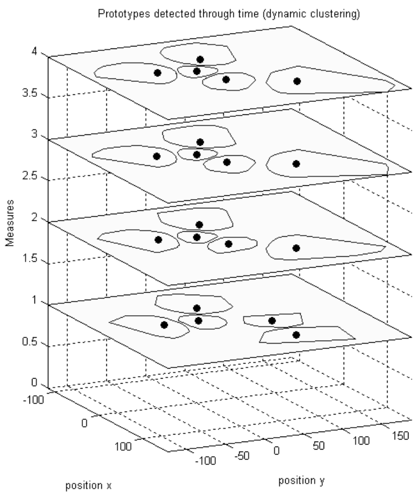
4.2 Results

In this case, the measures have been collected from all the agents at each cycle. In order to compare the results, in the previously described MAS (Fig. 3), the methodology presented has then been applied along with an ordinary clustering process (the FCM algorithm). The FCM algorithm is also the one applied at the dynamic mining methodology.

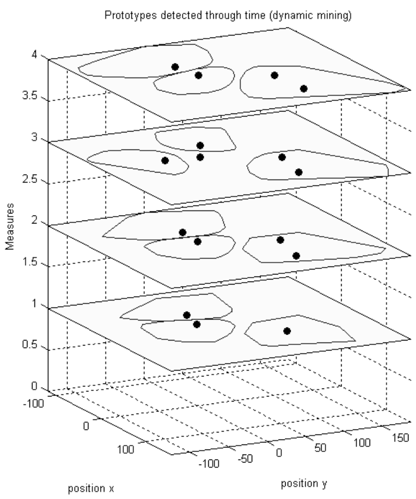
The results are displayed graphically in Fig. 4(a), with repetitive clustering on the data at each cycle (the data from one cycle at a time or *static* data); Fig. 4(b), with repetitive clustering on the stored available data from all the cycles (the *dynamic* data); and Fig. 4(c), for the methodology of dynamic data mining presented in this paper. The dots represent the centers of the clusters or prototypes, and the lines encircle their area of membership. It can be seen that the methodology of dynamic mining allows a flexibility that repetitive clustering might not reflect. As cycles count and the system evolves, the dynamic mining analysis adapts to the dynamics, giving a view of how the different patterns have changed through time.



(a)



(b)



(c)

Fig. 4. Comparison among static clustering (a), clustering on all the database form all the cycles (b), and the proposed methodology for dynamic data mining (c)

5. CONCLUSIONS AND FUTURE WORK

This paper presents a methodology to capture the different trends in a dynamic MAS, taking into account previous information about prototypes detected and how they can vary from an initial configuration. The methodology consists in a pattern detection and refinement process, developed in two steps: first, a Pattern Recognition System (see section 3.1.1), and second, a clustering procedure (section 3.1.2).

This research is still in progress, since, considering that this methodology effectively gives an overview of the dynamic of a MAS, its efficiency must be proved compared with other known methods. As can be seen from the simulation experiment, for instance, an application of a common clustering algorithm may yield similar or approximate results, with no measure of validity or adequation still available from both techniques. However, the ability of the detailed methodology to describe the dynamics of an evolving MAS at a glance, is considered as an appropriate starting point for further developments.

Identifying trends allows as well a further step, which consists in a prediction model. The information about the prototypes at each cycle can be used to train a model to predict the future outcomes of specific variables, or, in this case, the evolution of the clusters and the migration of their centers.

REFERENCES

Andritsos, P. (2002), Data clustering techniques, Qualifying oral examination paper, University of Toronto.

Andritsos, P. (2004), Scalable Clustering of Categorical Data and Applications, PhD thesis, University of Toronto.

Babuska, R. (1998), *Fuzzy Modeling for Control*, Ed. Kluwer Academic, Boston, USA.

Bar-Yam, Y. (1997), *Dynamics of Complex Systems*, Perseus Books Group.

Barbará, D., Li, Y. & Couto, J. (2002), Coolcat: an entropy-based algorithm for categorical clustering, *in* 'CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management', ACM Press, New York, NY, USA, pp. 582–589.

Bezdek, J. C. (1981), *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press.

Bonabeau, E. (2002), 'Agent-based modeling: Methods and techniques for simulating human systems', *Proceedings of the National Academy of Sciences (PNAS)* **99**(suppl. 3), 7280–7287.

Castro, A. & Oliveira, E. (2005), A multi-agent system for intelligent monitoring of airline operations, *in* 'Proceedings of the 2005 European Workshop on Multi Agent Systems (EUMAS)'.

Chan, K. C. C. & Au, W.-H. (1997), Mining fuzzy association rules, *in* 'CIKM '97: Proceedings of the sixth international conference on Information and knowledge management', ACM Press, New York, NY, USA, pp. 209–215.

Crespo, F. & Weber, R. (2005), 'A methodology for dynamic data mining based on fuzzy clustering', *Fuzzy Sets and Systems* **150**, 267–284.

Cruz, D., Mcclintock, J., Perteet, B., Orqueda, O., Cao, Y. & Fierro, R. (2007), 'Decentralized cooperative control - a multivehicle platform for research in networked embedded systems', *Control Systems Magazine, IEEE* **27**(3), 58–78.

Díez, J. L. (2003), Técnicas de agrupamiento para identificación y control por modelos locales, PhD thesis, Universidad Politécnica de Valencia. In Spanish.

Díez, J. L., Benítez, I. J. & Albertos, P. (2005), Dynamic analysis of diversity using multi agent systems, *in* I. García & F. J. Rodríguez, eds, '4th International workshop on Practical Applications of Agents and Multiagent Systems - IWPAAMS 2005', Universidad de León, pp. 169–172.

Díez, J. L. & Navarro, J. L. (1999), 'Fuzzy models of complex systems by means of clustering techniques', *Proc. 2nd Intelligent Systems in Control and Measurement* pp. 147–153.

Díez, J. L., Sala, A. & Navarro, J. L. (2006), 'Target-shaped possibilistic clustering applied to local-model identification', *Engineering Applications of Artificial Intelligence* **19**, 201–208.

Duda, R. O., Hart, P. E. & Stork, D. G. (2000), *Pattern Classification*, Ed. John Wiley & Sons, New York, USA.

Emami, M. R., Türksen, I. & Goldenberg, A. (1998), 'Development of a systematic methodology of fuzzy logic modeling', *IEEE Transactions on Fuzzy Systems* **6**, 346–366.

Ferber, J. (1999), *Multi-Agent Systems – An Introduction to Distributed Artificial Intelligence*, Addison Wesley.

Ferrari-Trecate, G., Buffa, A. & Gati, M. (2005), Analysis of coordination in multi-agent systems through partial difference equations. part ii: Nonlinear control, *in* '16th IFAC World Congress on Automatic Control', Prague.

Han, J. & Kamber, M. (2001), *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco.

Jain, A. K. & Dubes, R. C. (1988), *Algorithms for clustering data*, Prentice Hall, New Jersey.

Joentgen, A., Mikenina, L., Weber, R. & Zimmermann, H.-J. (1999), 'Dynamic fuzzy data analysis based on similarity between functions', *Fuzzy Sets and Systems* **105**, 81–90.

Karypis, G., Han, E.-H. S. & Kumar, V. (1999), 'Chameleon: Hierarchical clustering using dynamic modeling', *Computer* **32**(8), 68–75.

Krishnapuram, R. & Freg, C. (1992), 'Fitting an unknown number of lines and planes to image data through compatible cluster merging', *Pattern Recognition* **25**, 385–400.

Krishnapuram, R. & Keller, J. (1993), 'A possibilistic approach to clustering', *IEEE Transactions on Fuzzy Systems* **1**, 98–110.

Lutu, P. E. N. (2002), An integrated approach for scaling up classification and prediction algorithms for data mining, *in* 'SAICSIT '02: Proceedings of the 2002 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology', South African Institute for Computer Scientists and Information Technologists, Republic of South Africa, pp. 110–117.

Maravall, D. (1993), *Algoritmos de agrupación de clases: clustering*, Ed. RA-MA, Barcelona, Spain.

Martínez-Miranda, J., Aldea, A. & Bañares-Alcántara, R. (2003), Simulation of work teams using a multi-agent system, *in* 'AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems', ACM Press, New York, NY, USA, pp. 1064–1065.

Naso, D. & Turchiano, B. (2004), 'A coordination strategy for distributed multi-agent manufacturing systems', *International Journal of Production Research* **42**(12), 2497–2520. *http://dx.doi.org/10.1080/0020754042000197694

Nelles, O. (1999), *Nonlinear system identification with local linear neuro-fuzzy models*, Ed. Shaker Verlag, PhD dissertation, Darmstadt University of Technology, Darmstadt, Germany.

Ng, M. K. & Wong, J. C. (2002), 'Clustering categorical data sets using tabu search techniques', *Pattern Recognition* **35**, 2783–2790.

Nwana, H. S. (1996), 'Software agents: An overview', *Knowledge Engineering Review* **11**(3), 205–244.

Pereira, C. E. & Carro, L. (2007), 'Distributed real-time embedded systems: Recent advances, future trends and their impact on manufacturing plant control', *Annual Reviews in Control* **31**(1), 81–92.

Russell, S. & Norvig, P. (2003), *Artificial Intelligence - A Modern Approach*, Prentice Hall.

Sugeno, M. & Yasukawa, T. (1993), 'A fuzzy-logic-based approach to qualitative modeling', *IEEE Transactions on Fuzzy Systems* **1**, 7–31.

Wang, W., Yang, J. & Muntz, R. R. (1997), Sting: A statistical information grid approach to spatial data mining, *in* 'Proceedings of the 23rd International Conference onVery Large Data Bases (VLDB)', Morgan Kaufmann Publishers, Athens, Greece, pp. 186–195.

Zadeh, L. (1965), 'Fuzzy sets', *Journal of Information and Control* **8**, 338–353.

Zhang, Y. Q., Fraser, M. D., Gagliano, R. A. & Kandel, A. (2000), 'Granular neural networks for numerical-linguistic data fusion and knowledge discovery', *IEEE Transactions on Neural Networks* **11**(3), 658–667.