Applications of the Agents Reference Model for Intelligent Distributed Control Systems

JOSE AGUILAR
MARIELA CERRADA
HERMES DIAZ
FRANCISCO HIDROBO
GLORIA MOUSALLI
FRANCKLIN RIVAS

Universidad de Los Andes
Facultad de Ingeniería
Mérida, Venezuela.

Abstract:-In this paper we present the description of the control agents of a reference model for Intelligent Distributed Control Systems and an example to test the performance of our model in control problems. First, we describe reference model based on multiagent systems. Then, a case of study is presented.

1. Introduction

The works presented in [2, 3] define a hierarchical reference model to develop Intelligent Distributed Control Systems (IDCS). The hierarchical architecture is composed of a set of "objects " that execute the different tasks. The interaction of the objects consist on the messages exchange (commands, answers, etc.) between the different levels. This model has been extended using the Multiagent Systems theory (MAS), to integrate the different intelligence paradigms [3]. In this work we describe the control agents of our reference model for Intelligent Distributed Control Systems based on Multiagent Systems (IDCSA). Then, several examples of the control process domain are presented. Some of the problems are generics (for example, fault detection systems, supervisor systems, etc.), while others are more specifics (intelligent house control systems, car control systems, etc.). This work is divided in the following parts: In the next section the IDSC and IDSCA are presented. Then, we present the case of study to consider.

2. The Reference Model

In our model, the control tasks are distributed through the different hierarchical levels [2]. The hierarchy that we propose is composed of 5 levels; it is generally satisfactory for any control task. In general, this model can model a distributed control system that consists of a group of distributed entities which carry out control tasks. Our reference model can be used to model tasks of control of any nature. Our model can be used to solve the next control tasks: determination of the global state of the system, evaluation and selection of alternative, etc.

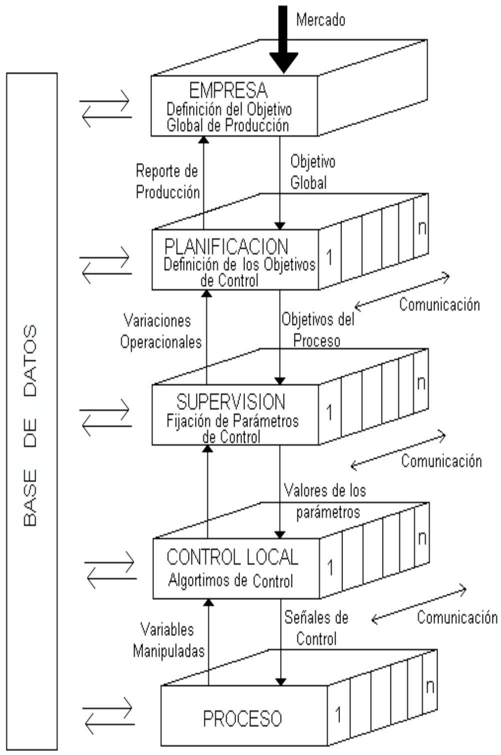

Figure 1. The IDSC Reference Model.

Our model incorporates intelligence through the hierarchical structure, in order to use different intelligence paradigms (specific reasoning mechanisms, different forms of knowledge storage, etc.) [3]. In this way, we must consider:

- A system to catch knowledge, which will be stored in the database. That is, a Knowledge Base with its motors of inferences, etc.
- A group of intelligent modules that are incorporated at each hierarchical level: Learning module, recognition Module, planning Module, adaptive Module.

In our model we can use different intelligent entities inside the layers or among the layers. In this way, we guarantee robustness, adaptativility, versatility and escalability to the system. This model is generic because it allows to unify different paradigms, tools and experiences in control.

We introduce the MAS theory in our model, in a way that the relationships among their elements emerges from the interactions of them [3]. Our model based on MAS defines a control system like a group of control agents that can be implemented individually. The IDCSA has characteristic of a decentralized system, with emergent, autonomous and concurrent properties. Particularly, the agents of our model have the following abilities:

- Learning: reinforcement, etc.
- Search: heuristic mechanisms, local or global searches.
- Inference or prediction.
- Representation of different sources of knowledge: natural language, communication language.
- Evolutionary.

These abilities are incorporate to the agents through the learning and adaptive module. In this way, each intelligent technique can be used. Their potentialities and virtues can be exploited efficiently.

2.1 Types of Generic Control Agents.

In the case of the agents for control systems, they could have some of the following functional levels:

a) Adaptation Level: it establishes the processes to carry out to reach their objectives. It manages a plan with the pending actions. It manages its knowledge base, beliefs and facts. Also, it generates a specific scheme of control composed by sub-tasks and actions. It detects conflicts and solves them through negotiation mechanisms.

b) Controller Level: it is the level that generates, executes and coordinates the realization of the actions according to its goals. Also, it assigns resources, and communicates with others.

These levels can have some of the four intelligent modules previously mentioned. In general, the IDCSA is a network of autonomous agents, with different responsibilities, according to the level that they belong. The agents are distributed through the control hierarchy and dispersed geographically. Each agent works in parallel, but they share and exchange information to cooperate [9, 10, 15]. These agents evolve for gradually to adapt her and improve her behavior. The objectives, communication, tasks, intelligence, etc., agent's characteristics are defined according to the level to which each agent belongs. For example, the agents of high level have tasks of coordination, planning, etc., while the agents of low level are strongly coupled to sensors and actuators through control mechanisms.

There are two forms of interaction between the agents: intra-levels and inter-levels. The complex agents can be seen as other MAS (composed by other agents), in this way, we define a hierarchy of agents. We will define agents' classes, with certain attributes that characterize them (abilities, limitations). According to IDCSA, the different types of generic agents to be used in the different layers, are the following:

a) Controller agents: they use the information sent by the agents to which are subordinate (commands, etc.) and transform them in control signs. Also, they generate and transmit commands from their level to the inferior level.

b) Actuator agents: they receive and execute control actions**.**

c) Coordinator agents: they make decisions, plan the tasks to carry out, etc. They control a set of agents. For the hierarchical structure of our IDCSA, they have different functions according to the level to which belong. For example, they fuse data and process information.

c) Mediator agents: they coordinate the communications between the agents. They make understandable the information send for the agents, etc. Also, these agents can solve conflicts.

d) Sensor agents: they collect information of the system.

e) Specialized agents: they carry out specific tasks of support to the system.

f) Interface agents: They interact with the environment, filter the information coming from the sensor agents, etc.

g) DB agent: It manages the database (DB).

h) Localization agents: They help to search agents in the system with certain required capacities. They are a support to the coordinator agent.

For more detail of this model to see [2, 3].

2.2 Generic Multi-agent Model for each level.

Our hierarchical reference model was developed based on the different control tasks that should be carried out in a process of any nature [3]. A generic multi-agent model for each level of our IDCSA has been described, based on the agents presented in the section 2.1 (see Figures 2):

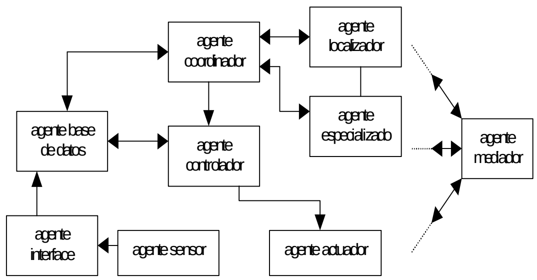

Figure 2. A generic multi-agent model for each level of our IDCSA

The collection of information and estimate of states are carried out by means of the sensor and interface agents, respectively. The evaluation and selection of alternative is made by the coordinator agent which will need of the localization and specialized agents in some cases. The definition of the control laws is carried out by the controller agent, and the execution of them by the actuator agent. The mediator agent coordinates and makes understandable the communication between the agents. The DB agent is the interface between the agents and the DB.

This model's architecture will depend on each problem. Each level will have the necessary agents to make up its tasks. For certain applications, we can carry out particular modifications in each level. In this way, we have defined a reference model of a distributed control compose by a group of distributed entities where they carry out control tasks.

3. Case of Study

3.1 A System of Water Distribution

In this example we define a control system for a system of water distribution for a city [8, 9]. The System to control consists of four tanks. The process of water distribution can be divided in three stages (see Figures 3). The first stage is composed by the first tank to which the liquid arrives. The second stage has two parallel tanks, which receive the liquid of the main tank. Finally, the fourth tank whose content comes from the three superior tanks. A part of the population of the city uses the water from the first tank. Other part uses the tanks from the second stage, and the rest of the population from the last one. The main objective of this system is to offer a distribution of water that fulfills the necessities of the whole population.

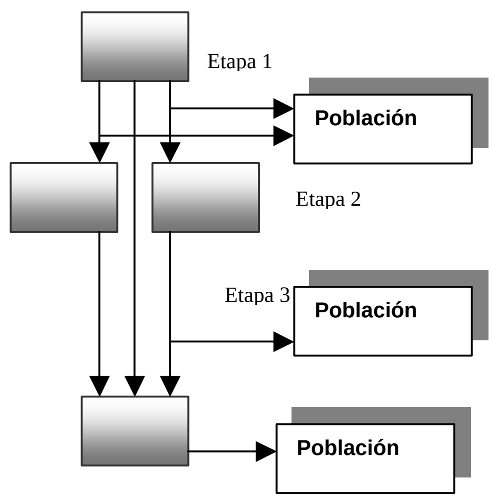

Figura 3. Red de Distribución de Agua

The control system will act over the speed of the bomb located to the exit of each tank to regulate the flow of liquid. That is, our objective is to design a control law to adjust the speed of the bombs in such a way of guaranteeing a good flow of water, maintaining the levels of the tanks according to the requirements of the own system.

According to our Model, we can define the following control scheme for this example (see Figures 4).

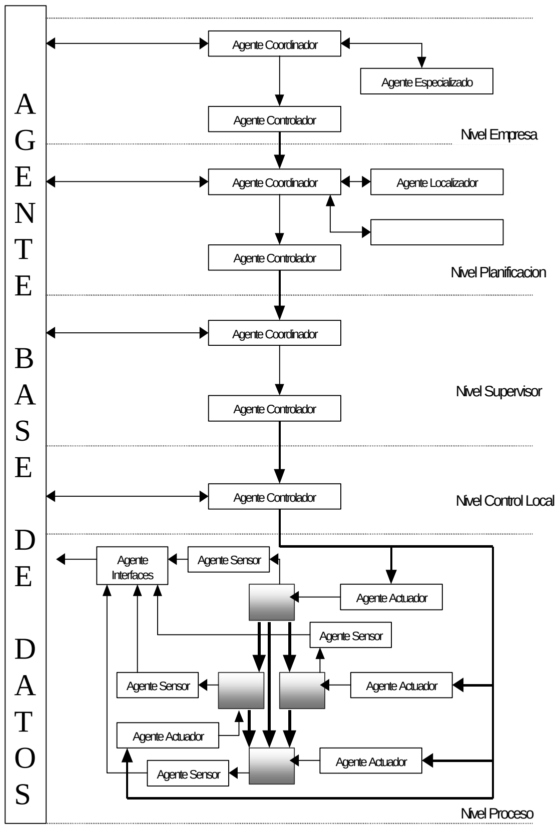

Figure 4. The tank control model using IDCSA

In our model, we have the following group of agents, defined by levels:

- Company Level: in this level, they carry out diverse tasks that allow to achieve the general objective of the company. The main goal is to provide water to the population with a constant flow that satisfies its necessities. This level requires the following group of agents:
- Coordinator agent: it coordinates the activities in this level. It can disable some of the activities of some agent.
- Specialized agent: it carries out specific tasks, for example: economic, or statistical analysis, etc.
- Controller Agent: it picks up the information coming from the coordinator and specialized agent, processes it and calculates the control parameters (quality parameters, environmental requirements, water flow requirements).
- Planning Level: it carries out certain tasks dedicated to the achievement of specific objectives. The required agents are the following:
- Coordinator agent: it plans the tasks to carry out in function of the objectives and requirements of the superior level. In this level, tasks of prediction, of definition of the good levels of the tanks, etc., are required. For some of these activities, it is necessary localization and specialized agents.
- Localization agent: it facilitates the work of the agents, it searches specialized agents according to specific requirements.
- Specialized agent: it carries out specific tasks for this level.
- Controller Agent: it picks up the data coming from the coordinator, generates the necessary information and sends it at the supervision level. It sents data correspond to specific control goals.
- Supervisor Level: in this level is carried out the adjustment or changes of the controller's parameters, in function of the execution of the control objectives. The agents required by this level are:
- Coordinator agent: it carries out the supervision of the system, takes the data coming from the process, manipulates them, and makes the decision about the modification of the parameters to control the process.

- Controller Agent: it calculates and sends the parameters from the controllers to the.
- Local control level: this level generates the control law.
- Controller Agent: it contains the set of control algorithms to guarantee a good flow of water. In this level, diverse control laws are generated to the actuadores of each tank. The parameters are adjusted depending on the information coming from the level supervisor.
- Process Level: this level is composed by the four tanks with its respective valves, bombs and pipes. In this level, is incorporated the sensor elements (they catch the information of the tanks) and the actuator elements (they modify the speed of the bomb). The agents used in this level are:
- Sensor agent: it picks up all the information of the process (level and speed of the bomb, etc.).
- Interface Agent: it receives the information coming from the sensors, it filters it and sends it to the DB agent. For each type of sensor agents there is an agent interface.
- Actuator Agent: it executes the control law. It acts directly on the valve, modifying their speed of pumping.
- DB Agent: it collects all the information coming from the different levels. It can be used as a form of indirect communication between the agents.

5. Conclusions

Our model is a powerful strategy to model and to design complex control systems. IDCSA allows the natural development of hybrid systems of control, where the intelligence can appear in different levels and in different ways. The transparency offered by IDCSA hides the possible heterogeneity of the different components of the system. Also, it provides a high modularity, extensibility, asynchronism, decentralized control and behavior opportunist. To use IDCSA is required to characterize the objects (functional units) that compose the system. Then, we must establish the relationships between the objects. That gives a wide autonomy to level of the design and implementation.

IDCSA allows to use the previous knowledge on the system, what is advantageous in systems where it is difficult to describe them under a precise mathematical model. Also, it allows to develop a design for aggregation. This can be seen at levels of the agents' definition, where each one of them can be another MAS. The inaccuracies and partial knowledge can be managed in our model. Also, the expansion of the control systems is easy (or the level of control detail), without costs in the execution speeds of the agents or in the operation of the levels (the efficiency of the system is protected).

We have presented an application, using an specific framework (for controlling the distribution of a liquid in a system of tanks). In this case we have not considered a specific technique to implement a given agent. The definition of each agent must be based according to the specific problem to solve.

Next studies must validate our model, through of real implementations of distributed control systems. For that, we must define the computational platform that supports our model of agents. One of the fundamental aspects of that platform is the mechanisms of the agents' interaction, taking into account the different elements that compose that interaction: commands (to coordinate and distribute tasks), data and knowledge. Another fundamental aspect is the connection of this platform with the agents, in such a way of guaranteeing the freedom and autonomy when implementing them. Finally, we also will study like the knowledge must be modeled in the agents and in the system (for example, we could use fuzzy cognitive maps [16]).

Acknowledgment:

This research was supported by the CONIPET (Venezuelan Council for Petroleum Research) under grant #97003817

References

[1] J. Aguilar, M. Cerrada, K. Morillo. "A Reliability-Based Failure Management Application Using Intelligent Hybrid Systems", *Proc. 4Th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes-SAFEPROCESS (Ed. A. Edelmayer)*, Vol. 1, pp. 197-202, 2000.

[2] J. Aguilar, M. Cerrada, F. Hidrobo, F. Rivas. "Esquemas Generales Basados en Sistemas Inteligentes para Control de Procesos: Conceptos Introductorios". *Primer informe de avance, Programa de CONIPET Agenda Petróleo*, Enero 2000.

[3] J. Aguilar, M. Cerrada, F. Hidrobo, F. Rivas. "Modelo de Referencia para el Desarrollo de Sistemas de Control Distribuido Inteligentes (SCDI)", *Segundo informe de avance, Programa de CONIPET Agenda Petróleo*, Agosto 2000.

[4] R. Aliev, R. Aliev. "Fuzzy Distributed System for Continuous Production in Application of Fuzzy Logic" *In "Towards High Machine Quotient Systems" (Eds. M. Jamshidi, A. Titli, L. Zadeh, S. Boverie)*, Prentice Hall, New Jersey, pp. 301-321, 1997.

[5] M. Cerrada, J. Aguilar. "Fuzzy Classifier System: An Application for Fault Tolerance in Industrial Processes", Coautor: M. Cerrada. *Proc. IASTED International Conference on Artificial Intelligence and Soft Computing (Ed. M. Hamza)*, pp. 453-456, 1998.

[6] M. Cerrada, J. Aguilar "RCM Methodology-Based Fault Tolerance Automated System", *In "Computational Mechanics: New Trends and Applications" (Eds. S. Idelshon, E. Oñate, E. Dvorkin)*, CIMNE e iacm, pp. 1-18, 1998.

[7] S. Debouk, S. Lafortune, D. Teneketzis. "A coordinated decentralized protocol for failure diagnosis of discrete event systems", *Proc. 4th Intl. Workshop on Discrete Event Systems*, pp. 138-143, 1998.

[8] B. Franek, C. Gaspar. "SMI++ object Oriented Framework for Designing and Implementing Distributed Control Systems", *IEEE Trans. on Nuclear Science*, Vol. 45, pp. 1946-1950, 1998.

[9] G. Haring, G. Kotsis, A. Puliafito, O. Tomarchio. "A Transparent Architecture for Agent Based Resource Management", *Proc. IEEE Intl. Conf. On Intelligent Engineering*, pp. 338-342, 1998.

[10] F. Hidrobo, J. Aguilar. "Resources Management System For Distributed

Platforms Based on Multi-Agent Systems", *Proc. 5th International Conference on Artificial Intelligence, Simulation and Planning-AIS' 2000*, pp. 60-69, 2000.

[11] S. McIlraith, G. Biwas, D. Clancy, V. Gupta. "Towards diagnosing hybrid systems", *Proc. 10th International Workshop on Principles of Diagnosis*, pp. 193-230, 1999.

[12] S. McIlraith, G. Biwas, D. Clancy, V. Gupta. "Hybrid Systems Diagnosis", *Proc. 3rd International Workshop on Hybrid Systems*, pp. 282-295, 2000.

[13] J. Mosterman, G. Biwas. "Diagnosis of Continuous Valued Systems in Transient Operating Regions", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 29, pp. 554-565, 1999.

[14] M. Passino, P. Antsaklis. "Fault Detection and Identification in Intelligent Restructurable Controller", *Journal of Intelligent and Robotic Systems*, Vol. 1, pp. 145-161, 1998.

[15] S. Shehory, S. Kraus. "Methods for Task Allocation Via Agent Coalition Formation", *Artificial Intelligence Journal*, Vol 101, pp. 165-200, 1998.

[16] C. Stylios, V. Georgopoulos, P. Groumpo. "Introducing the Theory of Fuzzy Cognitive Maps in Distributed Systems", *Proc. 12th IEEE International Symposium on Intelligent Control*, pp. 55-60, 1997.