AN AGILE AND COOPERATIVE ARCHITECTURE FOR DISTRIBUTED MANUFACTURING SYSTEMS

PAULO LEITÃO

Polytechnic Institute of Bragança, Quinta S^{ta} Apolónia, Apartado 134, P-5301-857 Bragança, Portugal, pleitao@ipb.pt

FRANCISCO RESTIVO

Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, P-4200-465 Porto, Portugal, fjr@fe.up.pt

ABSTRACT

World-wide competition among enterprises leads to the need for new systems to perform the control of distributed manufacturing systems, through the integration of information systems and self-organisation features, in order to adapt quickly to the environment changes. This paper presents an agile and cooperative architecture for distributed manufacturing systems based in the multi-agent technology which implements some holonic and bionic concepts and supports a new control approach, including the re-engineering phase.

KEY WORDS: Agile manufacturing, multi-agents, holonic manufacturing, virtual enterprises.

1. INTRODUCTION

The world-wide competition among enterprises, the organisation in temporary alliances (such as virtual enterprises), to allow industrial enterprises to keep their competitiveness, and the need to support new emergent manufacturing paradigms (such as mass customisation), lead to the need for new systems to perform the control of distributed manufacturing systems.

The new control systems for this new environment should face requirements that impose a capability of an agile and fast adaptation to the environments changes, through the integration of new technologies, tools and paradigms.

This paper presents a brief overview of the manufacturing paradigms evolution, as well as the requirements and problems associated to the new distributed and agile organisations. The research work on control architectures for manufacturing systems, both the traditional and advanced approaches, is also presented.

In order to fulfil those requirements it is presented a research work that intents to develop an agent-based architecture for agile and cooperative manufacturing systems, which supports the dynamic and agile adaptation to changes, the self-organisation and the use of standard interfaces with physical devices. The architecture implements some concepts derived from holonic and bionic manufacturing paradigms.

2. DISTRIBUTED MANUFACTURING ENVIRONMENT

2.1. Manufacturing Paradigms Evolution

During several years, the concept of mass manufacturing, characterised by the production of the same product in large scale, was widely implemented, but nowadays is incapable of treating variations in the type of product and can no longer respond to the challenges of modern and dynamic business. In the 80's, Japanese companies introduced the Lean Manufacturing, an extension of the Just in Time concept that aims to shorten the time line between customer order and shipment, by eliminating waste, and to achieve manufacturing products with less time to design, less inventory, less defects and reduction of setups. Another similar paradigm is the Agile Manufacturing, which consists in the ability to adapt quickly and profitably to continuous and unexpected changes in the manufacturing environment, and considers that human factors and organisational knowledge are just as important as advanced technology.

The CIM (Computer Integrated Manufacturing) paradigm, popular in the 80's, consists in the integration of the enterprise activities, through the use of information technologies, which allows the exchange and sharing of data among the various enterprise units. The main advantages are the increase of productivity, flexibility, quality, reduction of design time and work in progress [1], but due mainly to some technological, economical and social problems, the implementation of the CIM concept didn't have good results. Supply Chain Management deals with the management of materials, information and financial flows in a network, consisting of suppliers, manufacturers, distributors and customers.

The Virtual Enterprise is a paradigm that can be defined as a temporary alliance of enterprises that come together to share skills and resources in order to better respond to business opportunities [2]. The Extended Enterprise is a particular case, where an enterprise is preponderant in the network organisations.

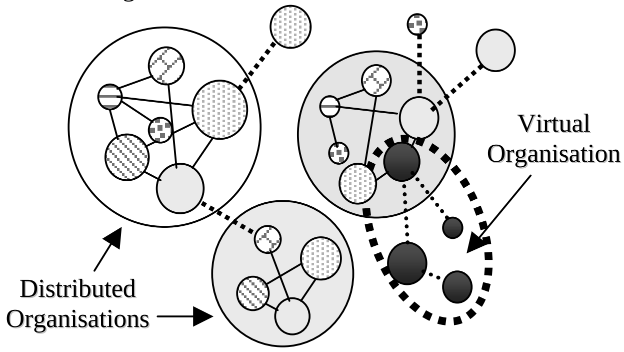

Figure 1 – New Manufacturing Organisational Structures

Nowadays, there is a trend for a high product customisation to fulfil the market demands, emerging the manufacturing from the era of mass production to the era of mass customisation. To accomplish this new environment, the companies are obliged to use new technologies and new manufacturing concepts, and combine them, in their different activities to avoid the risk to becoming less competitive or obsolete.

2.2. Requirements of Emergent Manufacturing Systems

The new ways of business organisation present requirements and problems, which increase the complexity of control applications. The main requirements that emergent manufacturing systems should comprise are [3]: enterprise integration, distributed organisational architectures, heterogeneous environments, integration of humans, co-operation, open and dynamic structure, dynamic organisation structure and fault tolerance.

On the other hand, they present important problems that are not completely solved, such as data translation, decentralised planning and scheduling and re-organisation techniques [4]. The data translation is related to the way to represent and understand the information in a decentralised environment is different for each company. The EDI (*Electronic Data Interchange*) and STEP (*Standard for the Exchange of Product model data*) protocols do not completely solve the data translation problem. Recently, the XML (eXtensible Markup Language) is pointed as the standard for the exchange of data. The decentralised planning and scheduling requires the synchronisation of local systems and processes supervision to support the global optimisation. Finally, the system should present self-organisation features in order to adapt to quickly to the external changes, but the way to represent the re-organisation techniques is a complex task.

3. EXISTING CONTROL ARCHITECTURES

In the research community, many control architectures have been proposed to fulfil the problems of advanced manufacturing systems, some of them using classic approaches and other using new and advanced approaches.

3.1. Classic Cell and Shop Floor Control Architectures

These architectures use mainly classic control architectures, such as hierarchical architecture. The COSIMA (Control Systems for Integrated Manufacturing) [5], is an architecture for cell and shop floor levels, which consists of five modules, grouped into a Production Activity Control (PAC): scheduler, dispatcher, monitor, producers and movers. The CHAMP (Chalmers Architecture and Methodology for Flexible Production) architecture was elaborated upon the experiences of PAC architecture and the main features are the separation of product and resource information, and the physical separation of generic and specific functions of the products and the resources currently in use [6]. Similar to those architectures, the Manufacturing Cell Controller of CIM Centre of Porto uses a modified hierarchical architecture approach. The interface between the cell controller and the physical devices is implemented using the MMS (Manufacturing Message Specification) protocol [7]. The MOSCOT (Modular Shop Control Toolkit for Flexible Manufacturing) object-oriented architecture is characterised by the kernel, which contains the common modules to all shop floor applications, and shells, which are developed or customised according to the specifications of each application [8].

3.2. Distributed Manufacturing control paradigms

The traditional manufacturing control systems have low capacity to adapt and react to the dynamic changes of its environment, and the new paradigms should have the capability to adapt to changes without external interventions. In order to achieve the above characteristics, several theories for the emergent manufacturing systems were proposed, presenting similar concepts and characteristics but with different origins: mathematics for the fractal factory [9], nature for bionic manufacturing (BMS) [10] and social organisation for holonic manufacturing (HMS) [11]. A comparative study of these theories can be found in [12].

3.3. Advanced Manufacturing Control Architectures

The new emergent manufacturing control architectures use mainly the multi-agent technology to support the development of autonomous and adaptative control architectures. The PROSA (Product-Resource-Order-Staff Architecture) is a holonic reference architecture based in three types of basic holons: Product, Order and Resource

[11]. Additionally, staff holons assist and advise the basic holons. The MetaMorph is a project developed at University of Calgary, using mediator centric federation architecture for intelligent manufacturing [13]. This research project evolved to other multi-agent architecture, which aims to integrate the activities of a manufacturing enterprise, with those of its suppliers, customers and partners within a distributed environment. The project proposes a hybrid agent-based architecture, intituled Agent-Based Manufacturing Enterprise Infrastructure (ABMEI), combining the mediator and the autonomous agents approaches [14].

The AARIA (Autonomous Agents at Rock Island Arsenal) architecture provides an integrated MRP and MES functionality in a hierarchical multi-agent approach and allows the dialog with customers and suppliers, to optimise schedules and to react to disturbances in the factory [15]. The MASCADA approach focuses on the manufacturing execution system, and is composed by communicating local intelligent autonomous agents [16]. The HOLOS/MASSIVE architecture supports multi-agent based dynamic scheduling and the integration with legacy systems [17]. This work was later on extended to distributed multi-site systems and virtual enterprises. The DEDEMAS (Decentralised Decision-Making and Scheduling) approach aims to provide mechanisms for decentralised decision-making and scheduling covering both multi-site operations of one company, supply chain and virtual enterprise cases [18]. Similar to HOLOS/MASSYVE, the decision-making mechanism is based on extended Contract Net, and several monitoring schemes and rules will support companies to optimise their processes. The GNOSIS-VF (The Virtual Factory Model-Based Distributed Manufacturing) project [19], concern about development of the Virtual Factory Platform, i.e. information technology support for the Virtual Factory. The main objectives are the design of the distributed co-ordination and decision organisation for the virtual factory, and the development of tools for virtual factory platform.

4. AGENT-BASED ARCHITECTURE FOR MANUFACTURING APPLICATIONS

In the section 2 the requirements of emergent manufacturing systems were presented and in section 3 some research in the control architectures to fulfil those requirements was analysed. In spite of the contribution of that research work, some issues remain with no efficient solutions to all requirements simultaneously. Additionally, nowadays the control architectures are focused in the operational phase and deal with the capability of adaptation to the environment changes.

However, little attention has been devoted to the changes of the manufacturing system itself along its life-cycle. A new approach should consider both the changes imposed by environment in which the manufacturing system is acting and the changes in the manufacturing system, during the re-engineering phase. To accomplish this requirement, it is defined a new control level that comprises the operation and re-engineering phases [20]. The architecture proposed supports the new control approach and tries to fulfil simultaneously all the requirements presented.

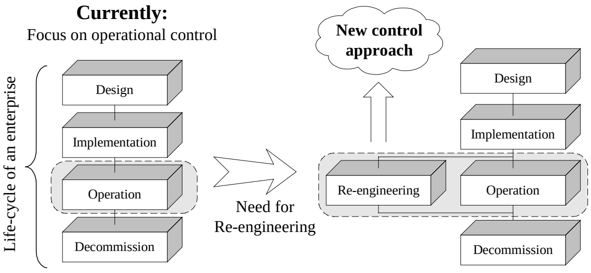

Figure 2 – New Control Approach Architecture

The multi-agent systems are suitable to the distributed manufacturing environment, since the manufacturing applications presents characteristics like modular, decentralised, changeable, ill-structured and complex, for what the agents are best suited to solve [21]. Analysing the benefits of multi-agent technology it is possible to conclude that fulfil some of requirements defined in section 2: autonomy (an agent can operate without the direct intervention of external entities, and has some kind of control over their behaviour), cooperation (the agents interact with other agents, in order to achieve a common goal), reactivity and proactivity (the agents perceive their environment and response quickly to changes that occur on it. In other hand, agents do not simply act in response to their environment, but are able to taking the initiative, controlling its behaviour) and adaptation and decentralisation (the agents can be organised in a decentralised structure, and easily can reorganised into different organisational structures).

4.1 Basic Concepts

The proposed architecture is a set of autonomous, intelligent and co-operative agents, forming a multi-agent platform. This agent-based approach implements the basic ideas of HMS, such as the possibility to represent a human and allow different organisational structures, and some ideas from BMS, such as the role of supervision and the dynamic evolution of the system. The agents can organise themselves in different organisational structures, in order to optimise the individual and community objectives, and combine the robustness and the reaction to the disturbances.

The architecture defines a set of agent classes: Operational, Supervisor, Product, Task and Interface agents. The Operational agent represents the interactions with the physical resources and the Supervisor agent has the objective to control several operational agents, depending of the organisational structure. The Interface agent allows the interaction with legacy system, users, etc. The Product agent represents the product knowledge and the associated information, such as the process plans. The Task agent is launched to represent the execution of a task in order to produce a product and contains the dynamic information about the manufacturing orders.

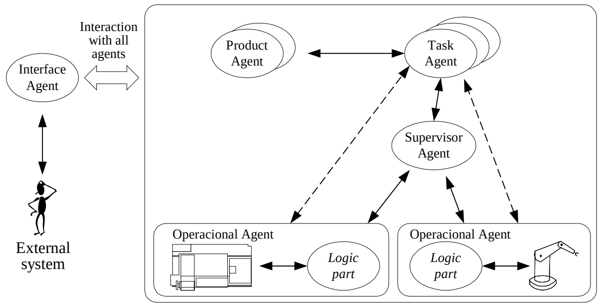

Figure 3 – Agent Classes in the Architecture

4.2 Organisation Structures

One of the main features of the architecture is the capacity of self-organisation, that allows the re-organisation of the agents into different organisational structures, with special attention to a completely heterarchical architecture (autonomous agent approach), and other more hierarchical (federation approach). However, it is possible to add dynamically, new organisational structures.

To support this agile adaptation to different organisational structures and in a further step the self-organisation, it is necessary to use methods to represent the relationships between the agents for each organisation structure and techniques to handle this feature. For example, if one agent left the system, it is necessary that all other related agents update automatically and dynamically their relationships.

4.3 Communication Mechanisms

The interaction between the agents in the architecture uses the TCP/IP protocol, is asynchronous and supports several communication modes, depending of the intention and knowledge of the agent: point-to-point, multicast and broadcast. The communication language is derived from KQML (Knowledge Query and Manipulation Language) language and the format of messages for data translation uses the XML language. Ontologies are used for the standardisation of the messages contents.

4.4 Disturbances Management

In the manufacturing environments there are disturbances that deviate the process from the original plans. Additionally, it is important to consider the shop floor re-engineering process that implies changes in the factory layout. In case of disturbances the system should respond dynamically and quickly, using mechanisms according the type of disturbance. The mechanism for disturbance management comprises a disturbance engine, which finds out the best plan to handle the disturbance, based in pre-defined rules and in the knowledge acquired with the experience.

In case of machine failure, the disturbance engine should decide, among other things, if the machine continue to be used in the reschedule or if it is necessary to reschedule all the production orders allocated to the machine, depending of the status of machine after the disturbance.

To handle the environment volatility (handle disturbance process) and improve his performance and abilities (handle decision process), it is important to implement a learning component in the agent-based architecture. The learning process used in the architecture, is based mainly in the acquiring knowledge based in the experience.

5. ARCHITECTURE FOR A GENERIC AGENT

The architecture for a generic agent, that belongs to the proposed architecture, is based in four modules and a local database, which contains all relevant information about the behaviour of the agent.

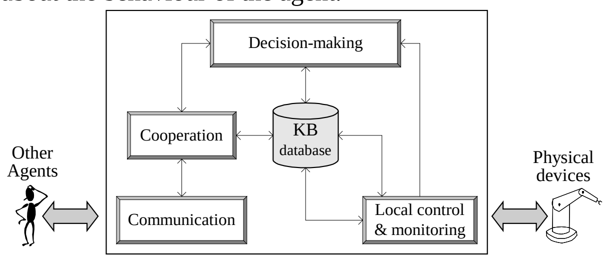

Figure 4 - Architecture for a Generic Agent

5.1 Decision-making module

This module controls all activities of the agent and includes the problem solving and decision-making. The decision-making process comprises the analyses of the available information, the application of the decision mechanisms, the evaluation of the decision and the learning. To support the decision-making, the module uses the knowledge stored in the local data base, and when the available information is not enough to make a decision, it is started a co-operation process with other agents trying to find out the necessary information to make the decision.

5.2 Co-operation module

This module manages the cooperation with the external agents, requesting cooperation to other agents, collecting the responses, and sending them to the decision-making module. This module considers and develops mechanisms for three different types of cooperation: negotiation (for product specification and order allocation), physical synchronisation and monitoring (passive and active forms). For the negotiation, this module implements a negotiation protocol based on the well-known Contract Net Protocol [22], extending the original functionalities including the following features: the negotiation process should allow multiple iterations, in order to have dynamic re-planning and iterative optimisation, and several ways of specifying the task announcements and bids, in order, to support the bid of partial quantities of the task. The cooperation module has the enough intelligence to reply a message, in case that the message is not adequate to the initial request. For the architecture proposed, it is not intend to develop complex mechanisms for cooperation and negotiation but focus on simple, practical, robust and reliable mechanisms, to support the dynamic adaptation and integration in agent-based platforms for manufacturing domain.

5.3 Communication module

This module deals with the need to standardise the interaction between distributed agents and defines a communication language. An additional aspect is the use of standard ontologies that defines the vocabulary that will be used in the communication between agents, and the knowledge relating to these terms. This module is divided in three main levels: contents, message and physical.

When the cooperation module wants to send a message to other agent it sends the information to send and the indication of the target agents. The contents level interprets the message and applies the appropriate ontology in order to standardize the data translation. The message level formats the message to send, using a language derived from the KQML language. The physical level allows the physical iteration with other agents and will use CORBA. The advantage to use the CORBA platform is the platform independency (interaction between applications running in distinct platforms, such as Windows, Linux and Unix) and language independency (interaction between applications developed in different programming languages, such as Java and C++).

5.4 Local Control and Monitoring module

This module intends to control and to monitor the operational execution of the agent. This module is divided in four sub-modules: control manager, dispatcher, monitor and operational control. The control manager aims to manage a task with several operations and the dispatcher aims to format and to dispatch the tasks to the physical devices. The monitor allows the passive and active monitoring of the resources activities. The operational control allows the communication with the physical devices or with the legacy systems.

5.5 Knowledge Base module

The local database stores all knowledge about the behaviour of the agent and the community where the agent belongs. The information stored in the local database involves several types of knowledge, such as constraints, objectives, procedures, rules and experience, and organisational structures and techniques.

The approach to the local database has two dimensions: individual-community information and generic-specific information. The individual part of information is related to the local data of the agent, such as local objectives, while the community data is related to the data concerning the system as a whole, such as system objectives. The generic part of the data is concerned to the common information to all agents and the specific part of data is related to the specific information for each agent.

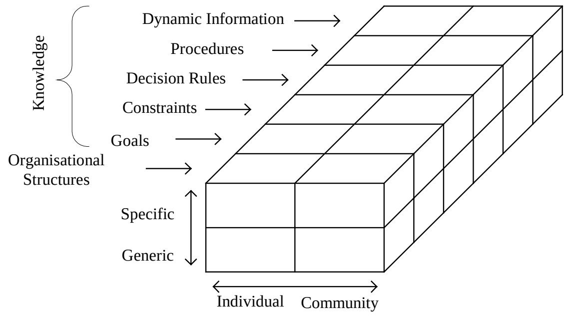

Figure 5 – Local Data Base for each Agent

The data stored in the local database, in a generic form, is concerned to the objects, the actors and the processes involved in the system domain application, such as the tasks and the resources.

5.6 Customisation of the Generic Architecture for an Agent

All agent types that belong to the architecture are based in the generic architecture of an agent. For example, the Operational agent uses all modules defined in the generic architecture, but other agents don't use all modules defined in the generic architecture, such as the Product agent that doesn't use the Local Control and Monitoring because it doesn't interact with physical devices.

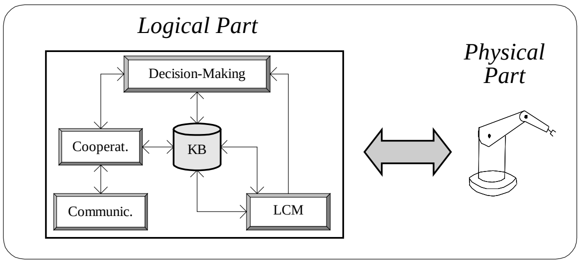

Figure 6 – Operational Agent

The Operational Agent uses an important concept derived from HMS: a holon can have both a logical (software) and physical component. In this way, the logical component controls the physical device through an appropriated communication system defined in the Local Control and Monitoring module. In order to standardise the interaction between the agents and the physical devices, this communication system implements a platform based in CORBA objects that defines the basic services for the control and supervision of the physical devices: write and read variables, program manipulation (download, start, stop, etc), etc.

6. CONCLUSIONS

Nowadays, the enterprises operate in a global and open environment characterised by dynamic changes in organisational structures and fast changes in technological and economical aspects.

The new requirements imposed by this new environment and by the market, require agile and dynamic control systems for distributed manufacturing systems based in the new technologies, tools and paradigms.

The paper reflects the research done in the distributed and cooperative manufacturing systems and the efforts to develop an architecture that fulfil the main requirements presented by this kind of applications, such as distribution of applications, cooperation, self-organisation, and integration of humans. In this paper it was presented an agent-based architecture approach to the development of distributed manufacturing applications that supports a new control approach, which includes the re-engineering phase, and uses some Holonic and Bionic Manufacturing Systems concepts and the multi-agent technology.

This is an ongoing work and therefore only preliminary results are presented.

7. REFERENCES

[1] Groover, M. P. *Automation, Production Systems and CIM* (Prentice-Hall, 1987).

[2] Camarinha-Matos, L. M. and H. Afsarmanesh, *Infrastructures for Virtual Enterprises: a summary of achievements*, Proceedings of PRO-VE'99 - IFIP Int. Conf. On Infrastructures for Virtual Enterprises, Kluwer Academic Publishers, ISBN 0-7923-8639-6, Porto, Portugal, 1999, 483-490.

[3] Shen, W. and D. Norrie, Agent-based systems for intelligent manufacturing: A state-of-the-art Survey, *Knowledge and information systems: an International Journal*, *1* (2), 1999, 129-156.

[4] Leitão, P. and F. Restivo, A framework for Distributed Manufacturing Applications, *Proceedings of ASI'2000 International Conference*, Bordeaux, France, 2000.

[5] Bauer, A., R. Bowden, J. Browne, J. Duggan and G. Lyons, *Shop Floor Control Systems - From Design to Implementation* (Chapman & Hall, 1991).

[6] Gullander, P., *On Reference Architectures for Development of Flexible Cell Control Systems* (PhD thesis, Gotenborg University, Sweden, 1999).

[7] Leitão, P. and A. Quintas, A Manufacturing Cell Controller Architecture, *Proceedings of Flexible Automation and Intelligent Manufacturing Conference*, Begell House Inc, ISBN:1-56700-089-4, Edited by William G. Sullivan and M. Munir Ahmad, Middlesbrough, 1997, 483-493.

[8] Teunis, G., P. Leitão and M. Madden, A New Architecture for Flexible Shop Control Systems, *Proceedings of Integration in Manufacturing Conference*, Gotengorg, 1998, 699-709.

[9] Warneke, H.J., *The Fractal Company* (Springer-Verlag, 1993).

[10] Okino, N., Bionic Manufacturing System, *J. Peklenik (ed.) CIRP, Flexible Manufacturing Systems Past-Present-Future*, 1993, 73-95.

[11] Bongaerts, L., *Integration of Scheduling and Control in Holonic Manufacturing Systems* (PhD thesis, Katholieke Universiteit Leuven, Belgium, 1998).

[12] Tharumarajah, A., A. Wells, L. Nemes, Comparison of the Bionic, fractal and holonic manufacturing systems concepts. *International Journal of Computer Integrated Manufacturing, Vol.9*, nº3, 1996, 217-226.

[13] Maturana, F. and D. Norrie, Multi-Agent Mediator Architecture for Distributed Manufacturing. *Journal of Intelligent Manufacturing, Vol. 7*, 1996, 257-270.

[14] Shen, W., Xue, D., Norrie, D., *"An Agent-Based Manufacturing Enterprise Infrastructure for Distributed Integrated Intelligent Manufacturing Systems"*, in proceedings of the Third International Conference on the Practical Application of Intelligent Agents and Multi-Agnets, London, UK, March, 23-25, 1997.

[15] Parunak, H. Van Dyke, A. Baker and S. Clark, The AARIA agent architecture: from manufacturing requirements to agent-based system design, *Workshop on Agent-based Manufacturing, ICAA'98,* Minneapolis, 1998.

[16] Heikkilä, T., M. Kollingbaum, P. Valckenaers and G.-J. Bluemink, manAge: An agent architecture for manufacturing control, *Proceedings of the 2nd International Workshop on Intelligent Manufacturing Systems*, KULeuven, 1999, 127-136.

[17] Rabelo, R. and L.M. Camarinha-Matos, Multi-Agent based Dynamic Scheduling, *Int. Journal on Robotics and Computer Integrated Manufacturing, Vol II*, n.4, 1994, 303-310.

[18] Tönshoff, H.-K., I. Seilonen, G. Teunis and P. Leitão, A Mediator-based approach for decentralised production planning, scheduling and monitoring, *Proc. of Int. Seminar on Intelligent Computation in Manufacturing Engineering*, Capri, Italy, 2000, 113-118.

[19] Gnosis-VF, http://www.vtt.fi/aut/tau/gnosis/

[20] Leitão, P., J. Barata, L.M. Camarinha-Matos, R. Boissier, Trends in Agile and Co-operative Manufacturing, *Submitted to Low Cost Automation 2001*, Dusseldorf, Germany, 8-9 October, 2001.

[21] Parunak, H. Van Dyke, *What can Agents do in Industry, and Why? An Overview of Industrially-Oriented R&D at CEC*. Industrial Technology Institute, CIA'98, 1998.

[22] Smith**,** R.G., The Contract Net Protocol: High-Level Communication and Control in a Distributed Solver, *IEEE Transactions on Computers, Vol. C-29*, Nº12, 1980, 1104-1113.