Journal of Achievements in Materials and Manufacturing Engineering
VOLUME 24
ISSUE 2
October
2007

Modelling of the manufacturing system objects interactions

J. Madejski*
Division of Materials Processing Technology, Management and Computer Techniques in Materials Science, Institute of Engineering Materials and Biomaterials, Silesian University of Technology, ul. Konarskiego 18a, 44-100 Gliwice, Poland
* Corresponding author: E-mail address: janusz.madejski@polsl.pl

Received 04.04.2007; published in revised form 01.10.2007

Analysis and modelling

ABSTRACT

Purpose: Modelling of the agent based systems is presented along with a brief description of the relevant software toolkit.

Design/methodology/approach: Areas of activity of the agent systems are presented, split into their internal and external domains, recognition of their world's state of affairs as well as planning of actions to affect their environment.

Findings: Overview of the approach to development of the agent based systems from the general goal notion to its constituent elements, like plans and tasks. The proposed hybrid control structure allows both the horizontal level negotiations among the local agents and the vertical intervention by supervisory agents; therefore, such attitude makes also possible the multi-task and many-to-many negotiations. The result is the global control and the possibility of the decentralized negotiation.

Research limitations/implications: Modelling LPA Chimera toolkit is based on Prolog which supports mainly backward chaining.

Originality/value: Selection of a powerful and flexible agent systems modeling tool has been made, exemplary analysis of the agent based system model design has been presented. There agents representing the physical system elements, like a humans, machine, production line, shop floor system, the entire plant, or simply a workpiece; it may also be the part-oriented scheduling, or even the complete scheduling process.

Keywords: Analysis and modelling; Artificial intelligence methods; Distributed artificial intelligence; CAMS; Holonic manufacturing systems

1. Introduction

Agent system design requires an efficient toolkit for modeling the future system. Among many available tools [1,2] the LPA-Prolog Chimera toolkit is one of the most useful for development of agents which can react to events and respond to them by posting further events. Creating an agent results in setting up a server staying in the background, monitoring for activity. The purpose of these servers is to be the components of a distributed computational environment. Such agent is a self-contained software module, which is capable both to offer services to other agents and request and take advantage of services of other agents – see Table 1. If another agent attempts a connection, this is acknowledged and completed without any need for programming or user interaction. In case the created agent desires to connect proactively to some other agent, the only information required to complete the operation is the IP address where the intended correspondent is running, together with a Port Number.

The particular agents are the intelligent nodes in a knowledge network, which can contribute to and assimilate knowledge, even from sources unknown at the time of their creation, as other agents – intelligent nodes - might have been created in the meantime. Cooperation of agents may be carried out on a local machine or across a network. Scheduling an event in another agent means that a term is posted to that agent. One can store any

LPA-Prolog data structure in the event terms, using numbers, variables, text, lists, etc. After establishing the communication channel, sending information between agents requires only calling a predicate to send the Prolog term to the other agent. The local agent may also schedule events by posting terms to itself.

Table 1.
Properties of agents [3-6]

Feature	Comment
Autonomy	An agent is capable of running itself autonomously, for the unspecified periods of time.
Awareness	An agent is aware of its environment, can detect other agents and interact with them.
Communicativity	An agent should can communicate with other agents, sharing its knowledge and offering its services.
Cooperation	An agent is able to cooperate with other agents, and may also help them to achieve their goals.
Knowledge	An agent has knowledge - set of "beliefs", which can be implemented a form of a database, blackboard, etc.
Proactivity	An agent is able to initiate operations, rather than simply respond to eternal requirements.
Purpose	An agent can have a long term goal justifying its existence.
Reactivity	An agent is modify its behavior in response to changes in its environment.

Posting implies sending an event to a single agent; whereas, broadcasting involves posting a message to multiple recipient agents. In case the list of agents which should be advised about some event has to be limited, then these agents which want to stay updated have to register.

Then certain events are broadcast to the selected connections only. All agents in the modeled system are allocated to their relevant tiers. A tier, or layer in some definitions, is a collection of agents with similar properties, whose definition may be more or less general. Obviously, agents with the more specific properties will be applicable for more specific tasks only, whereas those less concrete may be used more generally. The tiers form an architecture whose area of application is called its domain. Development of the agent system calls for design of its architecture, requiring analysis of the system domain from several viewpoints including: dynamic analysis, functional analysis, and information analysis. The agent system operates striving to achieve goals dynamically generated according to the changing agents' world perception (Fig. 1). Broadly speaking the agent subsystems may be split into the analysis of the current system state and planning and realization of the system goals which affect the environment using actuators carrying actions changing its world state of affairs.

All information about the environment and eventually is stored in the blackboard of the agent system in addition to the world perception of the system. This is the way in which they may broadcast messages to other agents and check the record of the environment state, as assessed by the processed sensors' data. It should be noted that the goals may be either only the representation of work or the actual work. The actual work to be carried out according to an instruction is called a task. These instructions may be sent among controllers or from a controller to the actuator, thus making it possible to carry out some action. Some exemplary goals and plans to achieve them are shown in Fig. 2. The particular plan steps are shown there, as generated by the planning tier. The steps are the elementary plan units, usually specifying at some conceptual level the single activities to be carried out [7].

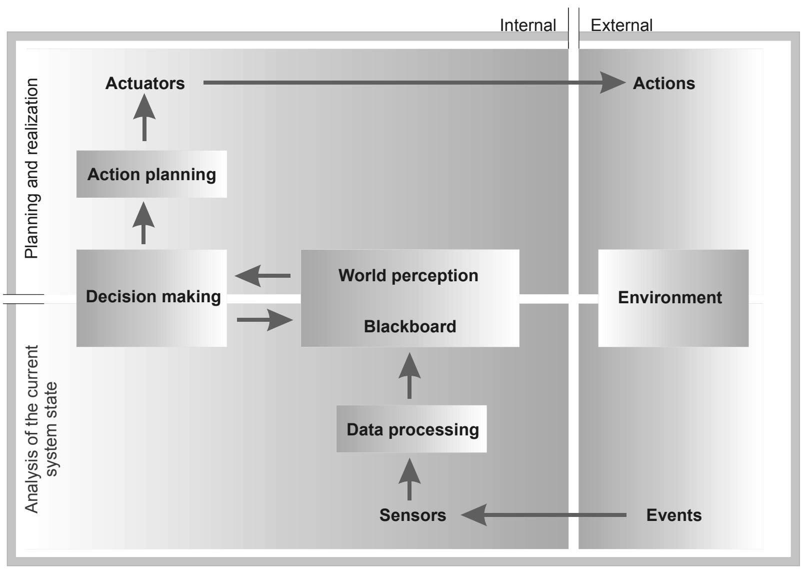

Fig. 1. General agent system structure

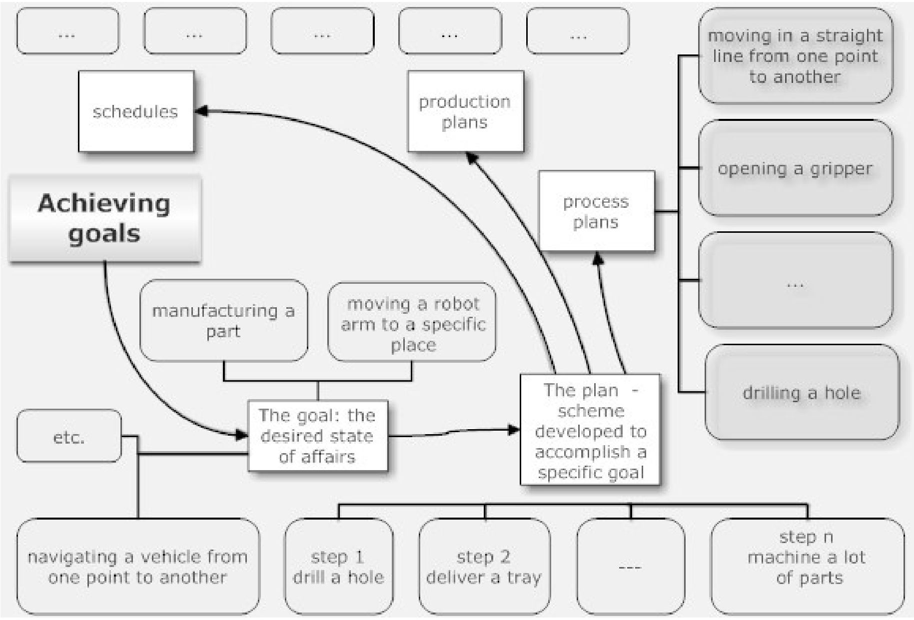

Fig. 2. Basic elements of the goal attainment process

2. System modeling approach

The most typical examples of tasks initiated by commands are the processing tasks. These are carried out by agents, therefore the system model should provide for dynamic creation of agents, as needed. The following LPA-Prolog Chimera command creates an agent, "agent1", using the predicate, "bar", as its event handler, at TCP/IP port "139" [8]:

```
|?-agent_create(agent1,bar,139).
```

Should this command be successful, the following command creates a connection between this and an agent running on a machine at IP address "192.168.1.4" and at port "583":

```
|?-agent_create(agent1,Link,`192.168.1.4`,583).
```

Agents created are maintained in a local database, and there is an access to the list of their names, which can be used for various informational and practical purposes.

Other types of activity, such as acquiring data, parsing the plan or synchronizing with other plans may also be carried out using the relevant commands [8-10]. It is expected that a command will name a work element and will provide the necessary values of parameters to the work element [11-13]. All tasks have to be allocated to the particular agents. Such allocation mechanism is usually set up by negotiations among agents using the contract-net protocol [14,15] consisting of five steps:

I. a requested task is announced by a manager,
II. potential contractors evaluate task announcements coming from many managers,
III. potential contractors bid upon some selected task,
IV. the task manager awards the contract to the selected bidding contractor,
V. the negotiation is completed by communication between the manager and the contractor regarding execution of the contracted task.

The advantage of this task allocation mechanism is that this algorithm is dynamic and easy to implement. However, its disadvantage is that any local decisions made by agents do not take into account the overall system performance. This protocol can be realized in the production system in the following ways [4,13]:

- managers and contractors are both resources; they can further subcontract tasks to distribute the workload,
- managers are the *product offering*, and contractors feature *resources bidding* to get the work assignment,
- managers feature *resources offering capacity* and contractors can be *parts bidding to use capacity*.

Three types of agents are used in the modeled agent systems. There agents representing the physical system elements, like a humans, machine, production line, shop floor system, the entire plant, or simply a workpiece; it may also be the part-oriented scheduling, or even the complete scheduling process. Agents of the next type are created on the fly to resolve scheduling conflicts. Finally there are agents serving as a supervisory entities focused the overall manufacturing goals. Anyway, an autonomous agent is expected here to be able to function without any human intervention and can decide if it needs to act in cooperation with other agents controlling its internal state and actions.

3. Conclusions

The proposed hybrid control structure allows both the horizontal level negotiations among the local agents and the vertical intervention by supervisory agents; therefore, such attitude makes also possible the multi-task and many-to-many negotiations. The result is the global control and the possibility of the decentralized negotiation.

The proposed LPA-Prolog Chimera toolkit, to be used as the simulation tool, makes implementation of the multi-agent manufacturing systems possible, consisting of both part agents and machine agents. Preliminary tests have proven that such hybrid approach is able to provide solutions with a good global performance. Further work will be concentrated on analyzing the effect of random disturbances affecting system operation. They will include machine and tool system breakdowns, changes of demand or product design. Therefore, developing the on-line protocol for rescheduling of system operation will be needed.

The final goal will be to develop a model of the holonic manufacturing with the system intelligence being distributed among its constituent entities - agents, called holons, which are the autonomous and cooperative [16-18]. Comparison will be made of the heterarchical approach banning all hierarchy to give full power to agents, with the hierarchical one. The simplest systems will, therefore, consist of workstations and orders only in which each order negotiates with the workstations to complete the work, attempting to face all unforeseen situations.

Comparison will be made of the efficiency of such systems with the heterogeneous ones in which resources are scarce and decisions at a certain time may have significant effects on future performance [19,20]. Attaining the enterprise integration will require each entity of the system to have access to the information it needs, according to its current task and making it understand in which way actions may affect other system entities.

The security problem will have to be analyzed because of the open architecture of agent based systems. This has become a serious issue in the advent of the mobile agent technology and the Internet.

References

[1] J. Madejski, Survey of the agent-based approach to intelligent manufacturing, Journal of Achievements in Materials and Manufacturing Engineering 21/1 (2007) 67-70.

[2] J. Madejski, Agents as building blocks of responsibility-based manufacturing systems, Journal of Materials Processing Technology 106 (2000) 219-222.

[3] J. Reaidy, P. Massotte, D. Diep, Comparison of negotiation protocols in dynamic agent-based manufacturing systems, Elsevier, International Journal of Production Economics 99 (2006) 117-130.

[4] T.N. Wong, C.W. Leung, K.L. Mak, R.Y.K. Fung, Dynamic shopfloor scheduling in multi-agent manufacturing systems, Expert Systems with Applications 31/3 (2006), 486-494.

[5] D. Wang, S.V. Nagalingam, G.C.I. Lin, Development of an agent-based Virtual CIM architecture for small to medium manufacturers, Robotics and Computer-Integrated Manufacturing 23/1 (2007) 1-16.

[6] M.D. Reimann, J. Sarkis, An Intelligent System for Automating the Inspection of Manufacturing Parts, in Design and Implementation of Intelligent Manufacturing systems. From Expert Systems, Neural; Networks, to Fuzzy Logic ed. by H.R. Parsaei, M. Jamshidi, Prentice Hall PTR, 1995, 19-38.

[7] F.P.M. Biemans, Manufacturing Planning and Control. A Reference Model, Elsevier, 1990.

[8] LPA Win-Prolog – Thinking Software, Tutorial, Logic Programming Associates Ltd, London, 2005.

[9] W.F. Clocksin, C.S. Mellish, Programming in Prolog, Springer 1994.

[10] P.H. Winston, Artificial Intelligence, Addison-Wesley Publishing Company, 1993.

[11] Ed. L.R. Nyman, Making Manufacturing Cells Work, Society of Manufacturing Engineers in cooperation with the Computer and Automated Systems, 1992.

[12] S. Ossowski, Co-ordination in Artificial Agent Societies. Social Structure and Its Implications for Autonomous Problem-Solving Agents, Springer 1999.

[13] J.P. Müller, The Design of Intelligent Agents. A Layered Approach, Springer, 1996.

[14] J. Madejski, The autonomous agent-based manufacturing systems architecture, Proceedings of the Scientific Conference on the Occasion of the 55th Anniversary of the Faculty of Mechanical Engineering of the Silesian University of Technology in Gliwice, Poland, M2E=2000, 293-302.

[15] J. Madejski, Fuzzy logic approach to the autonomous agent task utility function evaluation, Proceedings of the 8th International Conference Achievements in Mechanical and Materials Engineering AMME'99, Rydzyna, 1999.

[16] H. Van Brussel, L. Bongaerts, J. Wyns, P. Valckenaers, T. Van Ginderachter, A conceptual framework for holonic manufacturing: Identification of manufacturing holons, Journal of Manufacturing Systems, Journal of Manufacturing Systems, 1999 w http://findarticles.com/p/articles/mi_qa3685/is_199901/ai_n8841020/pg_1.

[17] B. Krupińska, D. Szewieczek, Computer assistance in the technological process efficiency analysis, Journal of Achievements in Materials and Manufacturing Engineering 20 (2007) 543-546.

[18] M. Musztyfaga, B. Skołud, Advisory system assisting selection of project structures and project team, Journal of Achievements in Materials and Manufacturing Engineering 20 (2007) 551-554.

[19] B. Krupińska, D. Szewieczek, Analysis of technological process on the basis of efficiency criterion, Journal of Achievements in Materials and Manufacturing Engineering 17 (2006) 421-424.

[20] A. Dobrzańska-Danikiewicz, D. Krenczyk, The selection of the production route in the assembly system, Journal of Achievements in Materials and Manufacturing Engineering 17 (2006) 417-420.