PROPOSAL FOR A DYNAMIC SCHEDULING ARCHITECTURE AND METHOD USING AN HOLONIC APPROACH

Nuno Silva, Paulo Sousa, Carlos Ramos

Departamento de Engenharia Informática
Instituto Superior de Engenharia do Porto
Instituto Politécnico do Porto
Rua de S. Tomé, s/n
4200 Porto - Portugal
Tel.: +351 2 8340500 Fax: +351 2 821159
e-mail: {nsilva, psousa, csr}@dei.isep.ipp.pt
http://www.dei.isep.ipp.pt/~{nsilva, psousa, csr}

Abstract: Manufacturing systems are changing its structure and organisation. Supply chain are evolving to more coupled organisations, like virtual enterprises, though maintaining the single entities autonomy, adaptability and dynamism properties. Such organisations are very different, which imply organisational and technological shift through agility, distribution, decentralisation, reactivity and flexibility. New organisational and technological paradigms are needed in order to reply to the modern manufacturing systems needs. This paper presents a holonic manufacturing system architecture and specially the scheduling sub-system along with a new scheduling method adapted from a traditional one.

Keywords: HMS, Multi-Agent, Scheduling and Architecture.

1. INTRODUCTION

During the last years, enterprises felt the need to change approaches to market reducing costs, time-to-market and product lifetime, at the same time increasing quality and environmental care. Hence, enterprise should be aware of concepts like concurrent engineering, virtual and federation enterprise, lean production, core business, etc. These new organisation concepts lead to new approaches in production and therefore in Manufacturing Systems. Distribution, Decentralisation, Co-operation, Co-ordination, Flexibility and Reactivity, are questions the manufacturing system should deal with more and more.

Scheduling in Manufacturing Systems is one of the manufacturing domains that must change in order to meet future organisation challenges. Scheduling corresponds to a distributed problem from the physical and from the logic point of view. Physically, the Manufacturing System involves several resources (numeric control machines robots, AGV's, conveyors). From the logical point of view it is also a distributed problem, because several tasks (task is meant to be a set of operations) can be carried out at the same time. Due to these reasons the framework of Distributed Artificial Intelligence (DAI) and Holonic Manufacturing Systems is proposed for dynamic scheduling of industrial tasks.

Section 2 describes some Multi-Agent and Holonic concepts and comparing them, help us explain why they were chosen as base paradigms for our project.

In section 3, the Dynamic Scheduling Manufacturing System Architecture is presented. In section 4, the scheduling method is described and proposed modifications are presented. In the last section, conclusions about already done work, and some remarks about questions not yet addressed are discussed.

2. MULTI-AGENT SYSTEMS AND HOLONIC MANUFACTURING SYSTEMS

Until few years ago, CIM concept was considered satisfactory enough treating enterprise/manufacturing requirements. However, taking in to account a new set of organisational and economic concepts, it becomes clear that the centralised CIM approach is not the answer. On the contrary, as stated in (Solberg and Kashyap, 93), these new concepts suggest autonomy, distribution, and flexibility, while stressing the need for co-operation among production units.

An agent is a rational entity sensing and acting on its environment in order to achieve its objectives. There are several properties needed by an agent, among others Autonomy, Social Ability, Reactivity, Pro-activeness (Franklin and Graessar, 96; Jennings and Wooldridge, 96; Jennings and Wooldridge, 95).

In addition to these properties, an agent can be grouped with other agents, thus forming a multi-agent. Not to be confused with multi-agent system which refers to society (or community) of agents, meaning there exist some sort of co-ordination/co-operation among agents to achieve a goal or goals. Although multi-agent paradigm responds to the new manufacturing systems challenges, it is too vague to satisfy design and development requirements.

The Holonic paradigm arises from Herbert Simon and Arthur Koestler studies about biological society evolution and organisation. Simon observed that complex systems are hierarchical systems formed by intermediate stable forms. These forms allow system to be stable, reliable and evolutionary, while maintaining a goal oriented functionality due to its hierarchical structure. He concluded that non-hierarchical systems are unnatural and inefficient and should be avoided in complex systems.

Later, analysing Simon theory and comparing it with its own observations, Koestler noticed that each system and its intermediate forms do not exist as auto-sufficient and non-interactive elements. On the contrary, they are simultaneously a part and a whole. A part since each element is always part of a bigger system, thus sub ordinate to the whole (upper levels), and a whole in the sense that each element is formed by multiple parts, thus supra ordinate its parts (lower levels) (Bongaerts *et al.*, 96). In order to designate these hybrid nature and behavioural entities, Koestler (1967) proposed the term Holon[1]. “A holarchy is a hierarchy of self-regulating holons, in supra-ordination to their parts, in sub-ordination to the higher levels and in co-ordination with environment” (Koestler, 67).

Additionally, the IMS – HMS group defined a set of properties related to the manufacturing systems based on the holonic paradigm:

- The holonic manufacturing system entities are autonomous and co-operative;
- Holon has information about itself and the environment;
- Each holon is composed by other holons and thus each holon is a also a holarchy;
- Each holon can dynamically belong to multiple holarchies;
- The holarchy has fixed rules and directives (the *canon* (Tharumarajah *et al.*, 96).

Each holon must refine the proposed tasks, co-operating with its lateral partners with whom must share a limited set of competencies, in order to achieve its own goals, defined by the higher holarchies. The co-operation process can be ruled by different methods like restriction propagation, negotiation (predefined and fixed conversation), speech acts (non-fixed conversation), voting (not very used), and master-slave (contradictory to autonomous entities).

The Holonic paradigm is an organisational and architectural concept since it defines the entity structures and its conceptual behaviours. The usefulness of the properties mentioned above requires implementation technology, which can be supplied by the multi-agent concept. Thus, holonic and multi-agent paradigms are complementary and not adversary.

The properties and behaviours defined for the holonic concept fits the structure and the dynamic behaviour of the modern manufacturing system requests, since it forms highly complex but dynamic and reactive structures, making no (or little) compromise with efficiency.

3. A DYNAMIC SCHEDULING MANUFACTURING SYSTEM ARCHITECTURE

As previously referred, a holonic manufacturing system is a holarchy composed by autonomous and co-operative entities, the holon, each one being

[1] *Holos* (whole in greek) + *on* (a particle, like in neutr*on*)

another holarchy. The proposed architecture, first described in (Sousa and Ramos, 96), follows this approach. It is a holarchy composed of multiple high-level holons (e.g. Process Planning, Scheduling, Stock Management) responsible for the co-ordination of its sub-entities. Lower, the product, the resource and the task holons (among others).

The adapted architecture model, presented in Figure 1, refers two higher level holons: Process Planning holon and Production Planning holon. During this case study, the focus is putted in the scheduling process. The scheduling activity relates its functionality with Production Planning and thus, the Scheduling holon sub ordinate to that holon.

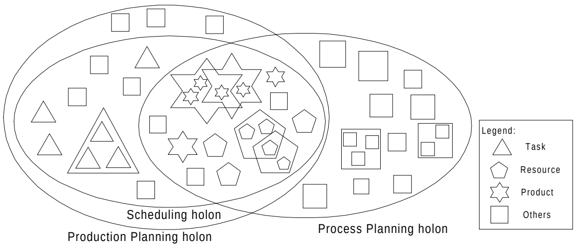

Figure 1 - Proposed architecture model (Silva, 98)

The Scheduling holon (like the activity itself) deals with several entities, both physical and logic:

- The Product holon represents the type of product, not the product object itself. The complete set of Products represents the system production possibilities (the catalogue). The Product is responsible for product planning specification and production quality control (Van Brussel *et al.*, 97). The Product can be composed by sub-Products, each one belonging to or be composed by others;
- Task holon models a production task received in the system, each one constrained by products, resources, quantity and quality parameters. The Task holon is responsible for scheduling, control, execution and conflict avoidance between entities performing the task (Van Brussel *et al.*, 97). The Task holon is potentially made of other Task holons, but contrary to Product, the Task can not belong to different Tasks;
- Resource holon represents the production elements in the system Each Resource can be made of others (e.g. a working cell) and share it with other (e.g. a tool used by multiple machines). The Resource holon is responsible for the operation control and sub-resources co-ordination, and concurs to the scheduling process, defining agendas, capabilities and capacities.

However, these entities also concern with other activities like the Process Planning. Consequently, some entities are simultaneously part of two holarchies, and thus, must contain and co-ordinate the functionality of the different processes they belong to:

- Resource holon simultaneously engages in product process definition and scheduling activities. First, supplying capacity and functionality information needed to process planning, second, supplying time agenda and activity control information fundamental to dynamic scheduling;
- Product holon, concurrently contribute to the product process re-planning together with the Process Planning holon, and responsible by the quality control together with Production Planning holon.

This of architecture is more flexible and reactive than the static and traditional CIM architecture. However, being a dynamic and non-deterministic system, security, information coherence and availability are relevant issues that system architecture design should concern with. Therefore, a few other entities were additionally combined into the system. Such entities, called Service holons are:

- Identification Service, responsible for identification, authentication and registration of holons in the holarchy, and co-ordination with other Identification Service holons;
- Information Service, responsible for the information management and co-ordination among multiples Information Service holons;
- Pooling Service, concerned with message delivery to holons temporarily unregistered in the holarchy. The service is requested by the holon that need to send messages to other;
- Routing Service is concerned with delivery messages to unknown location holons. The holon that does not know where its conversation partner is requests it.

4. SCHEDULING METHOD AND ALGORITHM

The requested scheduling system must be triggered when a new production order arrives. A Task holon is created and will be responsible for the negotiation between resources, each one represented by a holon. The task holon requests all the resources that can perform each task operation, and therefore, each resource can receive several requests and each operation can be "negotiated" by several resources. An n*m relation.

4.1. Original method

The original scheduling algorithm, developed for centralised manufacturing systems is described in (Ramos *et al.*, 95). Its main priority is to schedule tasks considering due dates. Briefly, for each operation, one single resource is requested and each one calculates a time interval list considering the previous operation time interval list respecting the time needed to perform the remaining operations until the due date. The time interval list calculated is passed to the resource requested for the next operation. Except the first resource, all the others must wait for the previous time interval list. This procedure is called the forward influence phase (Figure 2) and finishes when the resource is the last in the task plan. The time interval list found represents the last resource definitely availability to perform that operation.

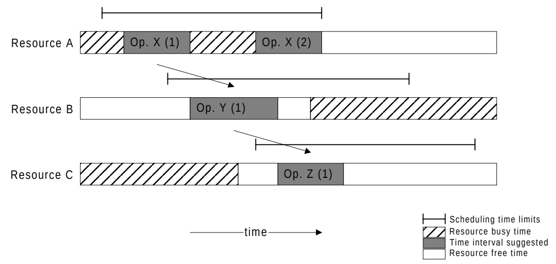

Figure 2 – Forward influence phase (Silva, 98).

Although this is the final list for the last operation, resources must influence previous resource intervals. Passing back its final list, resource informs the previous ones to readjust its suggested time intervals. This is called the backward influence phase (Figure 3) and finishes when the first resource readjusts its final list. Simultaneously with the backward influence phase, resource pass its list to the task holon, which, after receiving the list for all the operations, will initiate the time interval choosing process.

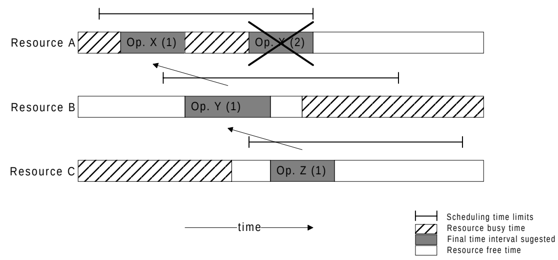

Figure 3 - Backward influence phase. Resource A reject 2nd interval because there is no further correspondence (Silva, 98).

Some questions must be highlighted:

- Each operation is executable in one single resource. On the contrary, in the conceptual system, any operation can be "negotiated" by multiple resources. If the original algorithm logic is followed, for each time interval list received, the resource calculates an influenced list and passes it to the next resources. Therefore combination explosion is possible to occur and should be avoided.
- The algorithm states that each resource must wait for the previous time interval list (back or forward). In requested system, is it reasonable to follow the same rule or not? Should each resource wait and process all the previous lists, or can it discard some list depending on some options and heuristics? If it discard, there is no guarantee that the "better" solution (or just the solution) will be found. On the other hand, if discard does not happen then the time to achieve the solution can be prohibitive in a few operations and resources.
- The solution is calculated from a time interval list. It is very simple and can be optimised to local and global policies. However, in the developing system, each operation is possible in different resources, which adds a new combination dimension to the problem, thus affecting choosing process efficiency and solution optimisation.

4.2. Proposed Method

To adapt the original algorithm to the holonic approach some modifications were proposed. Such modifications are based on assumption that each resource does not need to know what resource is suggesting the time interval. The method is divided in four different phases:

- Forward influence phase;
- Backward influence phase;
- Sequencing phase;
- Selection phase;

In the forward influence phase process the resource gathers the entire time spans in one single list, ignoring the resources identification. Next, each time span is compared with the resource free time. If the resource agenda allows the suggested interval to be used, the corresponding interval will be suggested to next resources. If the time span was no correspondence, it is discarded. This process is exemplified in Figure 4. Resource J and K, negotiating the execution of operation X, receive the same time interval list. In the example, both “answer” to one single time interval. Interval 3 is discarded since no resource has free time to execute the operation after interval 3 final time. Thus, to next resources, only two intervals are suggested.

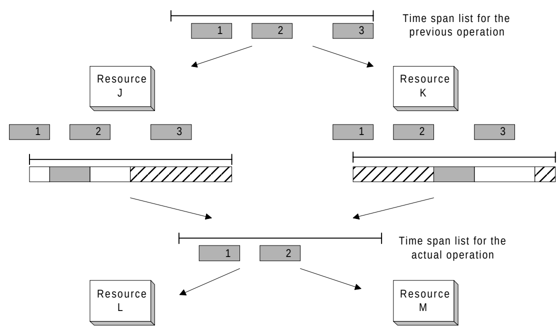

Figure 4 –Forward influence phase, representing the proposed changes (Silva, 98).

In the backward influence phase, the process is repeated: the resource sends only one list to all the previous resources. This list represents the definite resource availability to perform the operation. The resource validates its previously suggested time intervals comparing it with the time intervals received. Like in forward influence phase, if an interval is not corresponded, it is discarded and not suggested to the previous resources/operation. Each resource must send to the Task holon its definite time interval list for the operation. The example in Figure 5 represents the continuation of example in Figure 4. From the forward-suggested list, only interval 2 was further corresponded, which means Resource J will discard its interval. Thus, from the forward-suggested time interval list suggested from previous operation, only interval 2 will be acceptable.

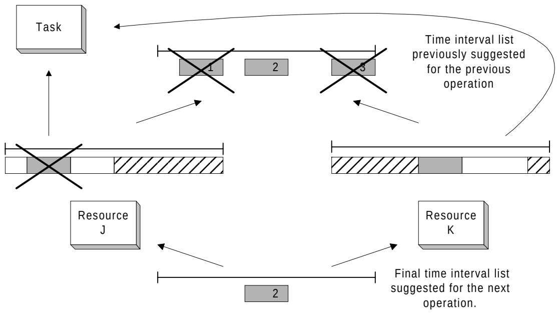

Figure 5 –Backward influence phase, representing the proposed changes (Silva, 98).

Since each resource receive several lists but sends only one, combination explosion is clearly avoided concerning message transfer. Consequently, each resource can wait for all the previous lists (forward or backward), that allows continuing find all the possible solutions. However, it is impossible to quantify the evolution of the number of time intervals treated during the scheduling process, since the intervals are highly dependable on the resources agendas, which are dynamic and unpredictable.

This question is very important in the choosing process, since the task must efficiently choose the better sequence of resource-interval. Notice that the problem already existed in the original algorithm, where some entity had to choose one of the different time intervals sequence suggested. The decentralised and negotiated approach adds one dimension to the problem: now there are different resources with different intervals for each operation, constituting more different possible sequences. In order to solve partially this problem, the method suggests the classification of intervals based on function defined by the Task concerning both local and global parameters.

During the backward influence phase the time intervals for each operation and resource are completely defined, therefore, the resource can classify each one considering a set of parameters or function, previously defined by the Task holon. Additionally, it is possible to use the resource local value to define an accumulated global value associate with the respective sequence. In that sense, the time span must be complemented with information concerning that accumulated value (Figure 6).

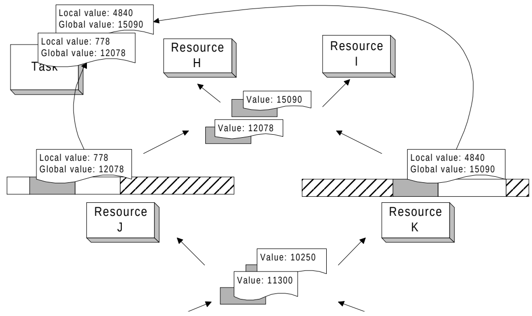

Figure 6 – Time interval classification with value propagation (Silva, 98).

Consequently, the Task holon knows for each time interval suggested the resource-operation value (local) and the global value. Each time span contains information about who had suggested it and related operation. The time span reaching the Task is arbitrarily ordered. However, it is possible to reorganise them to meet selection phase needs (Figure 7).

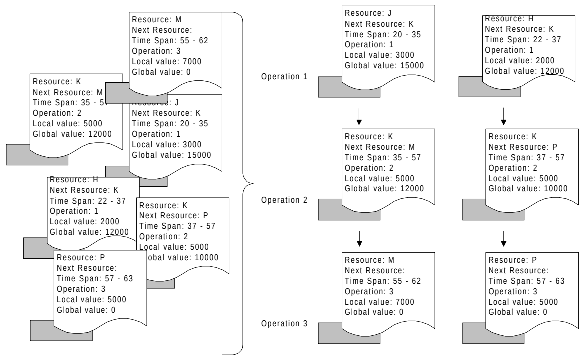

Figure 7 – Sequencing phase (Silva, 98).

Checking the global value received for the first operation, the Task holon knows the better sequence, based on the defined parameters and function. This value can be considered a cost if market based approach is used or some other unit depending on the used approach. It is possible the task to sort and filter in a more global forms the time intervals and thus more efficiently chooses the solution. In substitution or in addition to the classification method, some heuristics can be used.

5. CONCLUSIONS

This paper discusses the use of holonic approach to implement new generation of manufacturing systems. The holonic and multi-agent paradigms are briefly described and compared, which allows defending the opinion the two are compatible and even complementary, and not contradictory or opponent.

Presentation focus is putted in scheduling sub-system, defining the structure and entities functionality. The dynamic scheduling manufacturing system architecture is briefly presented and its principal entities and correspondent properties are highlighted. Four special services were presented.

Next, the original scheduling method is introduced. Due to different conceptual needs concerning both systems some problems are found in adapting it to holonic principles. Therefore, some modifications are proposed, most of them to address holonic properties, like distribution, decentralisation (multiples entities to execute the same operation), autonomy, co-operation and dynamism. Both the basic scheduling method as the previously described classification process, use propagation restrictions, which seems to be applicable and favourable.

Nevertheless, some issues are not completely study and implemented:

- Concurrent operations are very common in product plans, though the proposed method comport several limitations. In the original method this was not a problem, since there is a previous phase denominated Behaviour Generation that defines the plan operation sequence and parallelism. In the holonic approach this is not totally possible since the operation duration is not globally predefined in plan, but depends on particular resource, which are very dynamic during the system life-cycle.
- Heuristics used in choosing process. Although heuristics strongly depends on particular implementation and use, some studies should be done to validate method application.
- Due to distribution, time-out problems occur in resources conversations during time span propagation. Although this can be considered a control issue, the related problems affect considerable the scheduling method and are not easily solved or ignored.

By the moment, architecture and method are being implemented. Although no exhaustive tests were done yet, the preliminary tests are optimistic.

REFERENCES

Bongaerts *et al.*, 96; L. Bongaerts, J. Wyns, J. Detand, H. Van Brussel, P. Valckenaers; “Identification of Manufacturing Holons”; Pre-Proceedings of the European Workshop on Agent Oriented Systems in Manufacturing; Berlin, Germany, 1996.

Franklin and Graessar, 96; Stan Franklin, Art Graessar; Is it an Agent or Just a Program?: A Taxonomy for Autonomous Agents; Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996.

Jennings and Wooldridge, 95; Nick Jennings, Michael Wooldridge; Intelligent Agents: Theory and Practice; The Knowledge Engineering Review, vol. 10, pp. 115-152.

Jennings and Wooldridge, 96; Nick Jennings, Michael Wooldridge; Software Agents, IEEE Review, January 1996, pp. 17-20.

Koestler, 67; A. Koestler; The Ghost in the Machine; Hutchinson & Co, London, 1967.

Ramos *et al.*, 95; C. Ramos, Ana Almeida and Zita Vale, "Scheduling Manufacturing Tasks considering Due Dates: a new method based on Behaviors and Agendas", Int. Conf. on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Melbourne, Australia, 1995.

Silva, 98; Nuno Silva; Sistemas Holónicos de Produção – Especificação e Implementação; Dissertação de Mestrado; Faculdade de Engenharia da Universidade do Porto; September 1998.

Solberg and Kashyap, 93; J. Solberg, R. Kashyap; Research in Intelligent Manufacturing Systems; Proceedings of the IEEE, vol. 81, n. 1, January 1993.

Sousa and Ramos, 96; P. Sousa, C. Ramos; Scheduling Orders in Manufacturing Systems using a Holonic approach; European Workshop on Agent-Oriented Systems in Manufacturing; Berlim, Germany; September 1996.

Sousa and Ramos, 97; P. Sousa, C. Ramos; A Distributed Architecture and Negotiation Protocol for Scheduling in Manufacturing Systems; ASI'98, Budapest.

Tharumarajah *et al.*, 96; A. Tharumarajah, A. J. Welles e L. Nemes; Comparison of the bionic, fractal and holonic manufacturing system concepts; International Journal of Computer Integrated Manufacturing, 1996, vol. 9, nº 3, 217-226.

Van Brussel *et al.*, 97; Hendrik Van Brussel, Jo Wyns, Paul Valckenaers, Luc Bongaerts, Patrick Peeters; Reference Architecture for Holonic Manufacturing Systems: PROSA; Accepted for publication in Computers In Industry, special issue on intelligent manufacturing systems; 1997.