Myopia of service-oriented holonic manufacturing systems: the contribution of an observer

Olivier Cardin, Pierre Castagna

To cite this version:

Olivier Cardin, Pierre Castagna. Myopia of service-oriented holonic manufacturing systems: the contribution of an observer. Theodor Borangiu, André Thomas and Damien Trentesaux. Service Orientation in Holonic and Multi-Agent Manufacturing Control, SPRINGER, pp.197-210, 2012, Studies in Computational Intelligence, 978-3-642-27448-0. <hal-00784349>

HAL Id: hal-00784349

https://hal.archives-ouvertes.fr/hal-00784349

Submitted on 4 Feb 2013

Myopia of service oriented manufacturing systems: benefits of data centralization with a discrete-event observer

Olivier Cardin, Pierre Castagna

IRCCyN, 1 rue de la Noë, 44321 Nantes, France
IUT de Nantes, 2 avenue du Pr J. Rouxel, 44475 Carquefou, France

Corresponding Author : Olivier Cardin : olivier.cardin@irccyn.ec-nantes.fr;

Tel : +33 (0) 228092020 ; Fax : +33 (0) 228092021

Abstract

Service orientation paradigm is particularly well adapted to distributed manufacturing systems. The difficulty of such systems' production activity control deals with the knowledge management. Indeed, the knowledge is distributed among each entity, which is able to create, modify or communicate them with other entities. As a matter of fact, any entity cannot have a full up-to-date access to all the data of the system. On the shop floor level, a convenient way to implement service oriented manufacturing systems is to rely on the paradigm of Holonic Manufacturing Systems. This paper introduces the possibility of specializing a resource holon with the objectives to gather the data from the whole holarchy and make these data available to any holon for a decision making. This holon is thus playing the role of a discrete-event observer. After positioning the service-oriented architectures, the HMS reference architecture PROSA is described, especially in terms of decision making. After the decisions were defined, the problematic of online decision making in a HMS is described, and a solution of implementation of the observer and of forecasting tools in the architecture is exposed. Finally, two applications are presented, based on an industrial job-shop.

Keywords:

SOA, Service Oriented Manufacturing, Online simulation, HMS, Production Activity Control, PROSA, Staff holon, Observer.

1 Introduction

Since the beginning of the 21st century, service oriented architectures developed, dedicated to the interoperability of computer services in the companies. Along time, this architecture, mainly based on autonomy, negotiation and data distribution, was transposed on the shop floor [10]. In the field of systems involving a high level of information and control distribution, Holonic Manufacturing Systems are more and more common in both academic and industrial worlds [2], and particularly well adapted to the implementation of SOA. Many decisions must be taken by individual holons during a production. These decisions are based on the data that are retrieved on the HMS, but are generally insufficient for the operator to forecast the behavior of the system. In addition, the holons face the problem of myopia characteristic of HMS [1].

However, as shown in [13], efficiency of a holonic architecture goes by the ability for holons to forecast the future behavior of the underlying system. In this way, several authors suggest the use of ant colonies to predict the emergent behavior of the system on a short term [8, 12, 13].

The general framework of this paper deals with the use of discrete-event simulation as an online forecasting tool. This possibility was already exposed in [7] when saying that the use of tools supporting a discrete-event simulation of the production flows could ensure that no live-locks could occur when various products are considered at the same time in a product-driven system. Regarding their behavior, the underlying production systems can mainly be seen as discrete-event systems. As a matter of fact, discrete-event simulation is a very powerful tool to model HMS.

This paper introduces the possibility of specializing a resource holon with the objectives to gather the data from the whole holarchy and make these data available to any holon for a decision making. This holon is thus playing the role of a discrete-event observer. This work is based on a specific holonic architecture, named PROSA, and describes the architecture needed to integrate this observer, jointly with a forecasting solution in the holons called Staffs, based on online discrete-event simulation. After positioning service-oriented architectures in the first section, the HMS reference architecture PROSA is described, especially in terms of decision making. After the decisions were defined, the problematic of online decision making in a HMS is described, and a solution of implementation of the observer and of forecasting tools in the architecture is exposed. Finally, two applications are presented, based on an industrial job-shop.

2 Service oriented manufacturing systems

2.1 Service orientation in the context of distributed manufacturing systems

Service orientation paradigm is particularly well adapted to distributed manufacturing systems. In this context, the system can be seen as a set of multiple autonomous entities, which interact and cooperate in a complex network in order to accomplish a certain number of tasks.

The difficulty of such systems production activity control deals with the knowledge management. Indeed, the knowledge is distributed among each entity, which is able to create, modify or communicate them with other entities. As a matter of fact, any entity cannot have a full up-to-date access to all the data of the system.

This is why service oriented architectures (SOA) were suggested to handle the communication between the manufacturing entities. [9] defines the basic conceptual model of the SOA architecture, consisting of:

- service requesters: typically product entities when they are realized as order entities. Order entities call on the services they require to be manufactured,
- service providers: usually resource entities, which have the capabilities needed to provide the services that are requested,
- and service brokers: An actor that contains the rules and logics of using the services. Its function is to find service providers for the requesters on the basis of criteria such as cost, quality, and time.

2.2 Service oriented manufacturing system example

The studied HMS is a job-shop with automated transfers system is made of six workstations, each of them being a transformation service provider with its own intelligence[3]. The items are carried out one-by-one by a set of 40 transporters equipped with smart tags, all considered as separate resources, thus transportation service providers. A transporter storehouse (storage service provider) is available to store unused transporters.

When a new order is placed, a service requester is created, called here order service requester. A negotiation between this requester, the other main service requesters and the transportation service providers is established to determine how many transporters are dedicated to the order. When the decision is taken, the service requester is split into as many service requesters as items to be manufactured in the order, which are then called atomic service requesters. Then, each service requesters act independently: they are able to negotiate with each transformation service provider in order to go on its recipe, until the production of the item they represent is over, and the transportation service provider enters the storage.

Fig.1. The assembly line and the related decisions.

As a matter of fact, to run an order, the main service requester needs to make three decisions by negotiating with the other main service requesters (Fig.1 shows the localization of all these decisions on an assembly line scheme):

- DG1: At which date will the production begin?
- DG2: How many transporters will be allocated to this order?
- DG3: Which priority is given to the order?

The atomic service requesters also negotiate with the storage service provider to decide:

- DG4: Does the transporter located on the main loop at the entrance enter the storehouse or stay on the loop?

Along the production, atomic service requesters and transformation service providers negotiate to decide:

- DL1: Does the transporter located on the main loop at the entrance enter the station or stay on the loop? (particularly detailed in [4])
- DL2: At the end of an operation, is the atomic service requester authorized to step to the next operation of the recipe?

3 HMS and SOA

For the implementation of service-oriented architectures on a general point of view, multiple solutions are available. On the shop floor level, a convenient way is to rely on the paradigm of Holonic Manufacturing Systems. Indeed, the existing negotiation protocols, such as Contract Net Protocol (CNP) for example, are well adapted to the concept of service orientation.

The next sections introduce the reference architecture which was chosen for this implementation.

3.1 PROSA modeling

Many holonic architectures are exposed in the literature for the production activity control of HMS. One of the most famous, enabling a good compromise between hierarchy and heterarchy, is called PROSA [14]. As expressed in the acronym, PROSA (Product Resource Order Staff Approach) is mainly built around four kinds of holons (Fig.2).

Fig.2. Basic building blocks of a HMS and their relations, based on [11]

There are only two kinds of holons that are able to make decisions in a HMS: order holons (OH) and resource holons (RH). It is to be noticed that the type of decisions they make are completely different. Indeed, RH are inclined to make decisions on a very local point of view, when OH have to cope with the global objectives of the system, and therefore make decisions not only on a local, but also on a global point of view. Furthermore, as every holons deal with fixed rules and local variants, the mechanisms of decision making are quite different.

3.2 PROSA in SOA

In SOA, three main entities were identified: service providers, service requesters and service brokers. When looking carefully at PROSA, only Order and Resource Holons are making decisions. It seems also obvious to consider that Resource Holons might be identified to service providers, and Order Holons to service requesters. The service brokers are meant to support decisions. Their definition leads to an identification to Staff Holons (SH).

Of course, so that staff holons may be considered as services, they have to be accessible from different holons. Indeed, in the case of a staff holon only dedicated to the decision support of one only other holon, its definition as staff holon might be discussed. In this case, the question is to know why would not this staff holon be integrated inside the supported holon.

Finally, the Product holons (PH) remain. As stated in the Fig.3, it is generally considered that the process represents the service in itself in the context of service-oriented manufacturing systems. In PROSA, the Product Holon, representing the process data, is well adapted for this analogy.

Fig.3. An example of service in detail [9]

4 Centralizing data, not decisions

Online forecasting tools are numerous and have significantly different workings. However, the main problem for all these tools, identified in [5], is that the initialization of the tools must be made on the actual state of the HMS, as the decision horizon is generally short. But, it is very difficult to perform, as the data needed for this initialization are distributed among the holons. The first step is thus to have at anyone’s disposal a tool able to gather the state of the HMS up-to-date at any time.

The main contribution of this paper is to demonstrate that a specific holon, centralizing a huge amount of data coming from all the other holons and reconstructing the missing ones, would be very helpful in the production activity control of HMS as a support in the decision made by the other holons.

Next sections detail this proposition.

4.1 Gathering an up-to-date state of a HMS

The dynamic behavior of autonomous decisional entities, such as those found in holonic and multi-agent systems, makes it hard to obtain performance guarantees. This difficulty is mainly due to the ‘‘myopic behavior’’ of distributed control systems [15]. In fact, this myopic behavior is one of the major obstacles for using such systems. The analogy with myopia is justified since this condition causes a lack of visual acuity and can be extended to the lack of knowledge of a particular holon, about the whole system.

This section only focuses on the state gathering of the HMS. The objective of this work is to make a solution that could be applied to the widest class possible. As a matter of fact, the idea is to use as much as possible the real system’s data to have as reliable as possible data. However, the set of data obtained from the control is generally not sufficient. Therefore, the idea, which was given in [6], is to use an observer to reconstruct all the missing data.

Fig.4 presents the principles of such a solution. The observer is connected online with the whole holarchy so that it can retrieve all the events happening on the system. Then, its role is to continuously determine the missing data from the received ones. Using these data, it is also able to reset the deviation of its predictions. As a matter of fact, this observer is meant to run at the wall clock speed (real-time).

To sum up, the observer is meant to have three main specificities:

1. Gathering all the events happening on every other holons;
2. Reconstructing a probable behavior of each holon between two occurrences of events;
3. Make all or part of its state to any other holon which desire to gather data about the state of the holarchy.

A lot of technologies might be used to implement this observer, according to the objectives that were designed. To implement online simulation, the choice that should be made is to use discrete-event simulation. Indeed, a lot of simulation pieces of software meet the requirements of our study. First, it was widely used to model the behavior of such production systems. As a matter of fact, the model that was eventually made for the design of the facility can be used again in the production phase. This implies an interesting diminution of the investment time and costs. Then, the available means of communication are generally very well adapted to the communications inside such control architecture. Their graphical user interfaces is also very interesting, as it enables a clear vision on the behavior of the system for an operator. Finally, the state gathered on the observer is particularly well adapted to use to initialize online simulations in order to predict the future behavior of the system.

Fig.4. Use of an observer for gathering the state of a HMS.

4.2 Integration in HMS

As said before, the online simulation tools are best situated in a staff holon. However, this staff holon may be using several forecasting tools (ANN, Petri nets, etc.). These technologies might not be known from the other holons, which are only negotiating with the staff holon. Thus, it is necessary to split the staff holon into several resource holons, each of them representing a forecasting tool (Fig.5). In the most evolved decision making mechanisms, it is also possible to create order holons, representing the requests of the outer holons.

This organization is particularly interesting on the decision making delay point of view. Indeed, each forecasting tool having its proper delays, a negotiation with a short lap of time before the desired application of the decision might use a different tool than a negotiation with a longer delay.

The position of the observer is clearly as a resource holon. Indeed, its role is to deliver the state of the system to any holon at any time. The next question is to know whether this resource holon should be included in the staff holon. Considering the necessary link between this resource and the whole holarchy for retrieving the control data, it does not seem optimal to include this into a staff holon.

Fig.5. Integration of online simulation in a HMS.

Finally, with the observer considered as a resource outside of the staff, the placing of an order inside the staff is modified. Indeed, this means that one order's attribute is the considered state of the system, constant all along the forecast inside the staff holon.

5 Applications

This section introduces two examples, showing some of the benefits that can be encountered in terms of production activity control from the use of an observer coupled with a HMS.

5.1 Application to decision DG2

This section describes the use of an online simulation decision support tool on the HMS presented in section 2.2 to determine the best configuration of the orders placed on the system. This determination is made on the criteria of the total makespan. As a matter of fact, the evaluation also gives the information of the estimated makespan of the orders, running or to be run. The singularity of this decision is that it is finally made by the human in charge of the production activity control of the system. The forecasting tools are only used to provide data to this human, in order to help him make his decision.

The orders have several attributes, and among them:

- An ordered list of services needed for the completion of the item. This list is provided by the product holons ;
- A number of items to treat, divided into :
  - A number of transporters resources ;
  - A number of items each transporters has to treat ;
- A running date.

The difficulty is to assign the parameters related to the number of items. Indeed, several combinations are possible, each having positive and negative effects: for example, a low number of transporters increases the makespan avoiding the operations to happen simultaneously, whereas a high number of transporters decreases the makespan of this order, but only a few are left for the others (transporters are in finite capacity).

Fig.6 describes the communications between holons during a simulation-based evaluation of the total makespan on the occurrence of a new order with a given parameter in a sequence diagram. This sequence is repeated as many times as necessary to evaluate every desired parameter sets.

Fig.6. Sequence diagram of the makespan calculation.

After it was placed, the order triggers the calculation of the makespan according to the different scenarios. This calculation is limited in time by a duedate parameter, which prevent the staff holon to use too much time. Then, the staff holon gathers the state of the system, and creates an atomic order holon, corresponding to the request of the original holon. This order holon negotiates within the staff holon to obtain the data before due date. When decided, in this example in favor of online simulation, the replications are run. Finally, the results are propagated to the order holon.

5.2 Application to decision DL1

This section describes the use of the observer coupled with the HMS presented in section 2.2. This application deals with the use of the data available in the observer in the negotiations between holons.

The observer that was deployed has historical functionalities. Indeed, it is able to keep in memory specific events, such as items treated on each station. Then, it is possible to measure the load of each station, with the objective of balancing this load. This balance is made for maintenance purpose: if two stations are able to provide the same services, the maintenance costs tend to decrease when reaching a relative balance between these stations.

This section presents an academic study about the pertinence of using these data to balance the loads between several stations performing the same services

5.2.1 Problem definition

Table 1 presents the recipes of each item to handle. Table 2 presents the data relative to the services needed by the recipes. Finally, Table 3 presents the orders that are placed. By convention, the origin of the timeline (i.e. t=0) corresponds with the beginning of the manufacturing of these orders.

Recipe number	Service n°1	Service n°2	Service n°3	Service n°4	Service n°5	Service n°6
1	10	20	30	40	50	60
2	50	60	60	20	30	60

Table 1. Recipes used for DL1 scenarios

Service number	Setup Time	Manufacturing Time
10	30	5
20	60	20
30	10	4
40	160	10
50	10	30
60	**80**	**10**

Table 2. Services performed for DL1 scenarios

Order Number	Number of items	Number of items by transporter	Recipe
1	100	5	1
2	100	5	2

Table 3. Orders placed for DL1 scenarios

5.2.2 Scenarios

Three different scenarios were tested and are described in Table 4. First scenario is meant to be a reference scenario, where all the services are exclusively provided by 1 station. Difference between scenarios 2 and 3 is made by the presence of the load balance objective in the negotiation between the requesters and the providers.

5.2.3 Results and discussion

The results presented in Table 5 are expressed in seconds. The percentages relatively to the total makespan are not expressed to ease the reading of the data.

The results of scenario number 1 indicate that stations 1, 3 and 4 have a load strongly inferior to stations 2, 5 and 6. For the purpose of this study, a choice was made to try and balance the load of stations 1 and 6 on one side, and stations 3 and 5 on the other (Table 4).

Scenario Number	Load balance	Services performed by station					
		1	2	3	4	5	6
1	No	10	20	30	40	50	60
2	No	10,60	20	30,50	40	50	60
3	Yes	10,60	20	30,50	40	50	60

Table 4. Scenarios of DL1

This balance is first performed by simply declaring that stations 1 and 3 are respectively able to perform services 60 and 50 (Scenario number 2). Looking at the results of this scenario, the balance between stations 3 and 5 is globally acceptable, when the results of station 1 are not sufficient (still 4 times less load). Moreover, looking carefully at the results, it is possible to determine that the time saved on station 6 was almost integrally spent on station 1 for setups.

The difference between these balances can be explained by the topography of the HMS. Indeed, station 6 has a bigger buffer than station 5. Thus, very few items are not accepted by station 6 in the usual control, when a lot of items have to make supplementary loops on the central loop because of the lack of space in the buffer of station 5.

Performance indicator	Scenario Number		
	1	2	3
Load of station 1	530	1020	3460
Load of station 2	4060	4060	4060
Load of station 3	810	3970	3970
Load of station 4	1160	1160	1160
Load of station 5	6010	3820	3820
Load of station 6	4080	4030	3460
Cmax	6710	4784	4908

Table 5. Results of simulation

Thus, the decision was taken to implement a load balance decision rule in the negotiation between the service requesters and providers. This decision (Fig.7) is made after the provider (materialized by a resource holon) granted its access to the requester (materialized by an order holon).

Fig7. Sequence diagram of the load balance negotiation mechanism.

The chosen rule was the most simple: when the load of station 6 is greater than station 1, the items are not accepted in station 6. The results show that this rule is very efficient for the load balance (Scenario number 3). However, the Cmax indicator shows that some performance indicators are obviously degraded. This can be explained by the simplicity of the rule, which does not take into account anything else than the load balance.

6 Conclusion and future works

This paper highlights the benefits of data centralization instead of decision centralization for the production activity control of service oriented manufacturing systems, and more precisely their implementation as Holonic Manufacturing Systems. This data centralization is suggested to be performed using a discrete-event observer.

The concept of observer, enabling for example the forecasting tools to initialize on the actual present state of the system, was presented, and its integration in the HMS was explained. This observer can be seen as a specific resource of the holarchy, communicating with all the other holons to be able to retrieve all the data necessary to reconstruct the complete actual state of the HMS.

Furthermore, an example of the benefits that can be encountered in terms of production activity control from the use of an observer coupled with a HMS on an industrial FMS was developed, in order to exhibit the timed communications between all the holons involved in the production activity control (decision makers and decision supports). With very simple rules, a second example also shows how this data centralization can make the load balancing between several machines of a jobshop easy and efficient.

The future developments of this work will deal with the generalization of the approach to other control architecture, such as hybrid energy production systems. Indeed, the reconstruction of the state between two consecutive events is quite different, and makes the behavior of the observer harder to model.

7 References

[1] Adam E, Zambrano G, Pach C, Berger T, Trentesaux D (2011) Myopic Behaviour in Holonic Multiagent Systems for Distributed Control of FMS. Adv Soft Comp 90: 91–98

[2] Blanc P (2006) Holonic control of a laminated security glass production system. PhD Thesis. Université de Nantes, France.

[3] Cardin O (2007) Contribution of online simulation to production activity control decision support – Application to a flexible manufacturing system. PhD Thesis. Université de Nantes, France.

[4] Cardin O, Castagna P (2006) Handling uncertainty in production activity control. 12th IFAC Symposium on Information Control Problems in Manufacturing, INCOM 2006. Saint-Etienne, France. pp. 579-584

[5] Cardin O, Castagna P (2009) Using online simulation in Holonic manufacturing systems. Eng Appl Artif Intel 22: 1025–1033.

[6] Cardin O, Castagna P (2011) Proactive production activity control by online simulation. Int J Sim Process Model

[7] Gouyon D, Pétin J-F, Morel G. (2007) A product driven reconfigurable control for shop floor systems. Stud Inform Control 16.

[8] Karuna H, Valckenaers P, Saint-Germain B, Verstraete P, Zamfirescu CB, Van Brussel H (2005) Emergent forecasting using a stigmergy approach in manufacturing coordination and control. Lect Notes Artif Int.

[9] Nylund U, Andersson PH (2010) Simulation of service-oriented and distributed manufacturing systems. Robot Cim-Int Manuf 26: 622–628

[10] Trentesaux D (2009) Distributed control of production systems. Eng Appl Artif Intel 22: 971-978.

[11] Valckenaers P, Van Brussel H, Wyns J, Bongaerts L, Peeters P (1998) Designing Holonic manufacturing systems. Robot Cim-Int Manuf 14: 455–464

[12] Valckenaers P, Van Brussel H (2003) Holonic Manufacturing Execution Systems. Cirp Ann-Manuf Techn 54: 427–432

[13] Valckenaers P, Karuna H, Saint Germain B, Verstraete P, Van Brussel H (2006) Emergent short-term forecasting through ant colony engineering in coordination and control systems. Adv Eng Inform 20: 261-278

[14] Van Brussel H, Wyns J, Valckenaers P, Bongaerts L, Peeters P (1998) Reference architecture for holonic manufacturing systems: PROSA. Comput Ind 37: 255-274

[15] Zambrano G, Pach C, Adam E, Berger T, Trentesaux D (2011) Myopic Behaviour in Heterarchical Control of FMS. International Conference on Industrial Engineering and Systems Management IESM 2011, Metz, France.