V CONGRESSO BRASILEIRO DE ENGENHARIA DE FABRICAÇÃO
14 a 17 de abril de 2009 - Belo Horizonte - Minas Gerais - Brasil

HOLONIC MANUFACTURING EXECUTION SYSTEMS FOR CUSTOMISED AND AGILE PRODUCTION: MANUFACTURING PLANT SIMULATION

Jean Marcelo Simão, jeansimao@utfpr.edu.br [1]
Roni Fabio Banaszewski, ronifabio@gmail.com
Cesar Augusto Tacla, tacla@cpgei.cefetpr.br [2]
Paulo Cézar Stadzisz, stadzisz@utfpr.edu.br [2]

[1] Departamento de Eletrônica (DAELN) e [2] Departamento de Informática (DAINF),
Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial (CPGEI).
Universidade Tecnológica Federal do Paraná (UTFPR)
Av. Sete de Setembro, 3165 CEP-80230-901 (LSIP/CPGEI) Curitiba- PR Brazil.

Abstract: *The e-mass-customisation is one of the current tendencies in production, in which each client details desired products using e-means. This imposes advances on integration and collaboration of manufacturing entities, aiming at adaptability to achieve production heterogeneity and agility. An integration element of management and shop floor systems is the Manufacturing Execution System (MES). It may take into account smart-product technology to deal with customised manufacturing. In this technology, the execution of each order is driven by an entity called smart-product. A smart-product requests and even competes for services of resources, which in turn collaborate based on their features and some flexible logic. However, this collaboration is by itself complex, firstly due to the heterogeneity of the resources. Thus, resources and even smart-products have been "encapsulated" in collaborative entities called Holons (HLs), for homogenisation and integration issues. This contributes for achieving Holonic MES (HMES). HMES comprises also other issues like control of the Holon dynamics. In this context, authors' previous studies proposed a particular solution, based on "Rules" and "Notification", for control of Holon collaborations. This solution was developed as a HMES meta-model and implemented over a simulator called ANALYTICE II, applied mainly on simulated manufacturing cells. In turn, this paper proposes a meta-model application over a simulated manufacturing plant using ANALYTICE II. The proposed case study takes into account aspects of agility in customised production.*

Keywords*:* Holonic system, mass-customisation, product-driven control, H-MES, manufacturing-plant simulation.

1. INTRODUCTION

The mass customisation tendency is a challenge to the Intelligent Manufacturing System community (Da Silveira et al., 2001). In fact, customers want to buy products that meet their needs and desires using the easiness of the technology (e.g. e-commerce) (Gouyon et al., 2004). This requires production agility to customize the products in limited time. This agility can be partially achieved through Manufacturing Execution System (MES) whose purpose is the integration and synergy of management systems and shop floor systems (Morel et al., 2003)(Qiu et al., 2003). However, current MES technologies do not effectively allow customised production because they do not provide support for customisation of each single product, product traceability, re-configurability of resources, and so on (Simão et al., 2006). Thus, it is necessary to employ additional concepts like smart-product and Holonic Manufacturing System (McFarlane et al., 2003).

The smart-product concept defines each product instance coupled with a smart-entity that drives its production. Thus, a smart-product requests services to Manufacturing System resources (e.g. equipment and work cells) for achieving its production needs. In this context, resources are also improved with some collaborative smartness, in order to suitably deal with received requests in an agile manner (McFarlane et al., 2003). This allows production and order-dispatching independence, as well as the consistence maintenance of physical and informational flows.

Nevertheless, the collaboration of these entities is itself a problem mainly to their heterogeneity. The entity heterogeneity may be attenuated by their encapsulation in communicating entities with common interfaces called Holons (HLs). Thus, Smart-Product-HLs collaborate with Resource-HLs composing a Holonic Manufacturing System, which is by definition agile and able to deal with high-production variety (Van Brussel et al., 1998). However, Holonic

Manufacturing System composition is not trivial because the Holon dynamic is in general complex. Indeed, this subject motivated authors' previous studies where the organisation of Holon collaborations is achieved via entities called "Rules", which allow establishing a flexible collaboration logic (Simão et Stadzisz, 2002).

Rules decide moments for collaboration of Resource-HLs, based on state notifications from Resource-HLs, which allows requesting their services (Simão, 2005)(Simão et Stadzisz, 2008). In fact, Rules are allocated by Smart-Product-HLs that need operations related to the Resource-HL collaboration. Thus, the Rule set is a decoupling and organisational mechanism of Smart-Product-HL and Resource-HL collaborations (Simão, 2005)(Simão et al., 2006).

This solution represents a Holonic Manufacturing System (MES) evolution, which has been proposed as a control meta-model for Holonic MES (HMES) and tested in a Holonic Manufacturing System design and simulation tool called ANALYTICE II (Simão, 2005). However, the majority of tests were related just to simulated manufacturing-cells. Therefore, this paper proposes an alternative case study, which is related to a manufacturing plant and was redesigned/simulated in ANALYTICE II by using the proposed control meta-model.

In fact, a short-term objective of the authors' researches is to evolve the meta-model into an engineering tool to aid in different types of Holonic Manufacturing System (HMS) composition, by using a set of previously developed and tested smart-entities and their relationship. The meta-model must allow reducing the composition time of HMS of different types or levels, such as manufacturing cells and plants. This becomes a reality in ANALYTICE II where experiments allow demonstrating the potential of this approach.

This paper is structured as follows: section 2 presents mass customisation issues, HMS/HMES rationales, and the HMS design/simulation tool called ANALYTICE II; section 3 presents a process-driven control solution for HMES/HMS and describes previous experiments; section 4 presents the evolution of the control solution as a product-driven solution and describes other previous experiments; section 5 presents the application of the solution in a case study related to a manufacturing plant; and section 6 presents conclusion and future works.

2. HMS/HMES FOR MASS CUSTOMISATION

In order to deal with varying production, manufacturing organisations must exploit their own production flexibilities. Thus, some researches propose auto-organised Manufacturing System entities for improving the manufacturing processes (Deen, 2003)(Morel et Grabot, 2003). Classically, production is planned in lots, via Enterprise Requirement Planning (ERP) systems, based on previous client demands, where entities (e.g. controllers and machines) are prepared to produce few types of products in a period of time. This policy is not appropriate to mass customisation production once response time to product may be very long.

A solution is each production-order, related to a given product, to be a smart-entity that knows skills and states of advanced resources (i.e. flexible, configurable, and integrated resources) and that suitably launches its own production. The resources are also enhanced with some expertise to allow smart-orders knowing their states/skills and requesting their services (McFarlane et al., 2003). This may be accomplished by attaching an agent to each resource, via computational-electronic means as represented in Fig. (1). The physical resource and its agent are together considered as a smart-resource (Simão et al., 2006).

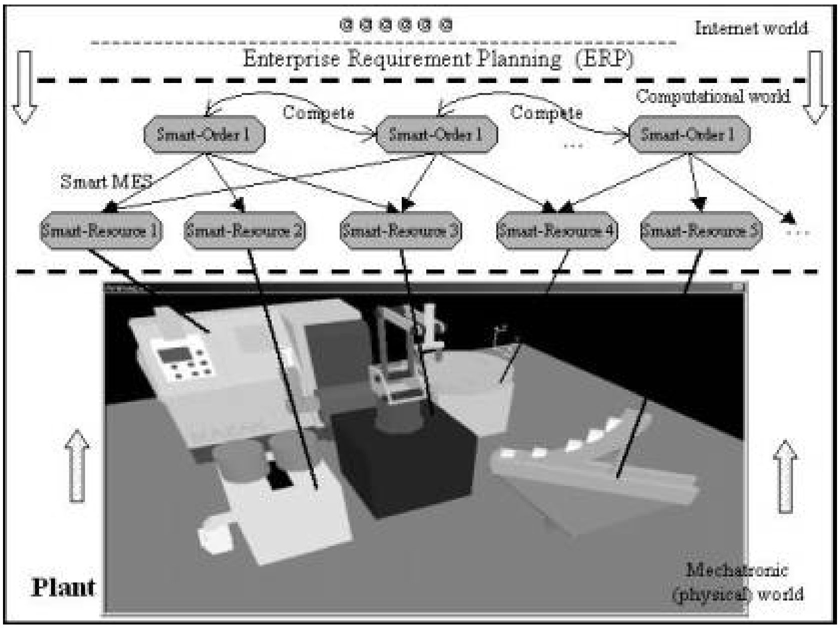


Figure 1. Integrated environment with Smart-Resources and Smart-Orders.

This entire scenario composes a "smart Manufacturing System" in which smart-orders negotiate with smart-resources and launch their own production. This order-driven approach makes collaborations easier once production actors and their negotiations are carried out at software level. However, some complicated questions remain, such as the possible following incoherency: a smart-order may believe that its concerned product is in a given place but it is not. A solution to this problem is each smart-order to be integrated with its product, thereby composing a smart-product. A manner is to identify the product with a given identification frequency and update the correspondent agent, which can be made via RFID (Radio Frequency Identification) technology (McFarlane et al., 2003).

The smart entity concept can be related to the Holon concept. A Holon is an autonomous and collaborative Manufacturing System building block for transporting, storing, and/or validating information and physical objects. It consists of an information processing part and often a physical part (Van Brussel et al., 1998). In fact, the most focused approach in the Intelligent Manufacturing System researches is the holonic paradigm, originated from a philosophical theory on the creation/evolution of world adaptive systems (e.g. social systems). The main idea is to achieve good properties of natural systems (e.g. adaptability and flexibility) in Manufacturing System (Morel et Grabot, 2003)(Valckenaers, 2001).

A Holonic Manufacturing System (HMS) is based on Holon collaborations, namely on Smart-Product-HLs and Resource-HLs. However, just Smart-Product-HLs and Resource-HLs negotiating in heterarchical manner (or free manner) are not enough once problems may appear, e.g. states unpredictability, deadlock or combinatorial explosion of states (McFarlane et al., 2003). Thus, a MES-like (i.e. a control system) is necessary to control their collaborations ensuring operability/adaptability by avoiding strong hierarchism, i.e. a heterarchy and hierarchy trade off, forming a holarchy. In fact, this MES-like is an industrial and Intelligent Manufacturing System-community concern (Qiu et al., 2003)(Morel et al., 2003)(Van Brussel et al., 1998).

The next sections present efforts in a Holonic MES (HMES) solution, which is validated over the singular HMS design/simulation tool ANALYTICE II. This tool was developed at LSIP/UTFPR initially for CIM (Computer Integrated Manufacturing) issues. Nevertheless, its building blocks allowed its holonification (Simão, 2005). Like in real Manufacturing System, ANALYTICE II separates the execution of resources and Shop Floor Control (SFC, a MES synonymous) via a 'virtual' network sketched in Fig. (2). Thus, an agent in the control side receives signals and requests services for each resource, via the network, forming realistic Resource-HLs. Besides, in ANALYTICE II or even in real Manufacturing System, Resource-HLs substitute a Shop Floor Control part usually called Supervisory Control and Data Acquisition (SCADA).

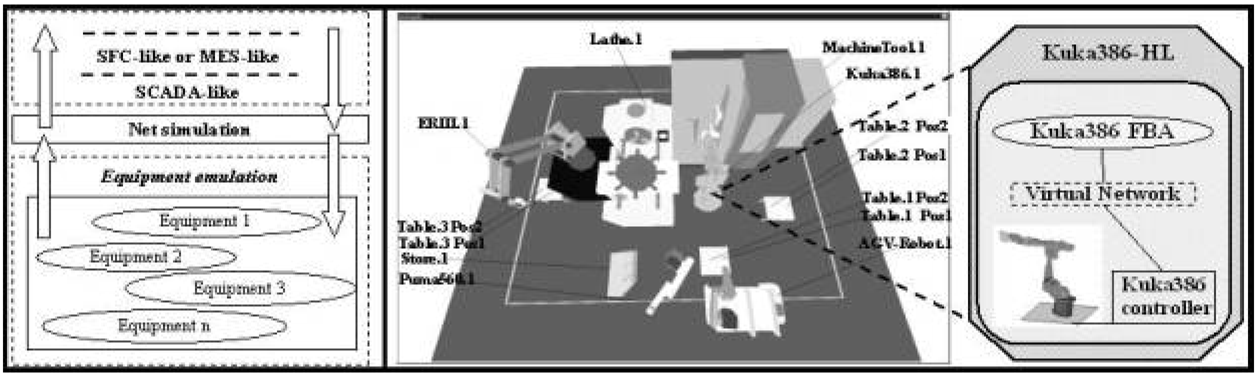


Figure 2. ANALYTICE II structure, its graphics animator module, and its holonification

3. HOLONIC CONTROL SOLUTION – PROCESS-DRIVEN CONTROL

The solution of HMES or Holonic Control firstly comprised the improvement of the Resource-HL concept. In this solution, Resource-HLs express resource states by Attribute (sub) Holons and receive services demands via Method (sub) Holons, homogenising their interaction manner and then facilitating control activities. The first MESs composed over instances of this type of Resource-HLs were process-driven controls (i.e. without smart-product use) that allowed generating a control architecture (Simão et Stadzisz, 2002).

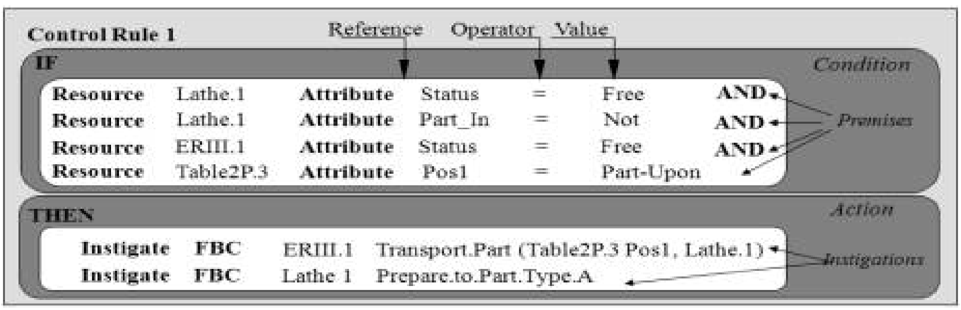


Figure 3. A Rule Holon and its associated (sub) Holons in causal-rule knowledge form.

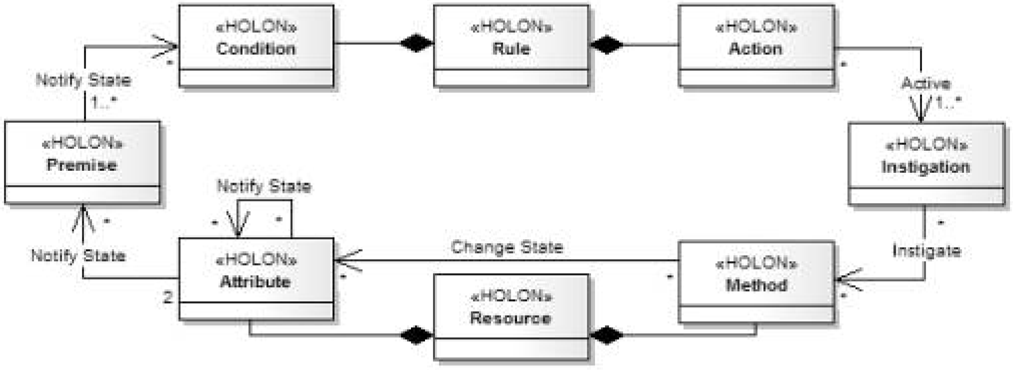


Figure 4. UML class-diagram of the Rule Holon (class) and its associated (sub) Holons (classes).

The solution was inspired by Rule Base System from which each instance is a type of Expert System to carry out the control of Resource-HL collaborations (Simão et al., 2003). In fact, each control instance is an Expert System-like whose fact-base is related to the states of Resource-HL Attributes, the decision & coordination is "carried out" by causal rules, and the final conclusion is a set of instigations of Resource-HL Methods.

An example of causal-rule that evaluates Attributes and instigates Methods of Resource-HLs is presented in Fig. (3). However, in this Holonic Control solution, the causal rules are not just simple passive data in a database. The causal-rules are Holons (called Rules) and so are their own associated entities (e.g. Premises and Instigations) (Simão et al., 2006), which constitute a differenced control and inference solution (Simão et Stadzisz, 2008). Thus, Fig. (3) presents in fact an example of Rule (Holon) knowledge.

In turn, the Fig. (4) presents a class-diagram of this Holonic Control solution in UML (Unified Modeling Language), where the Holon relations (based on notification) are modelled. This Holonic Control solution allows alternative inference by an elegant notification chain presented in Fig. (5). In short, the Attributes of Resource-HLs notify just related Rules about changes of their states via Premises and Conditions. This brings a set of control and inference advantages as detailed in Simão (2005) and Simão et Stadzisz (2008)[1].

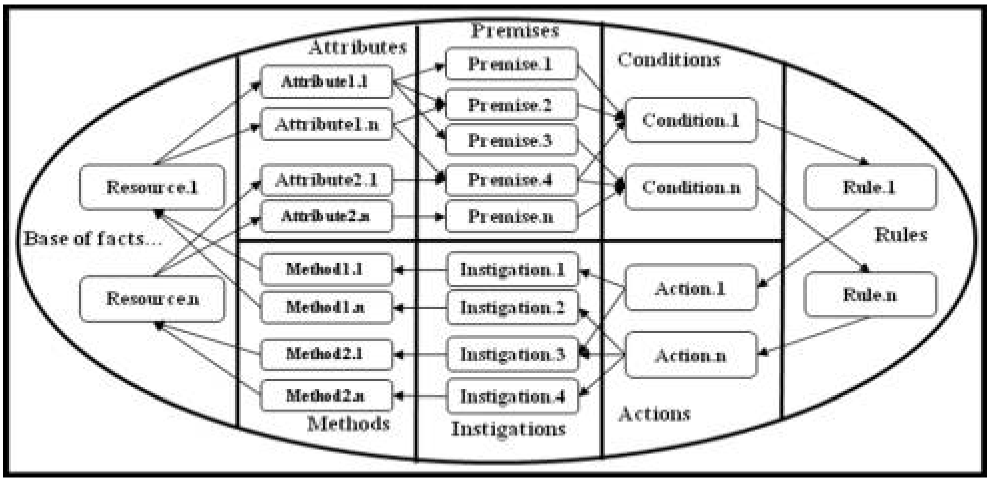


Figure 5. (a) Notification chain composed by the Holon for inference execution.

This Holonic Control solution was tested in a set of HMES experiments (Simão, 2005). Among the developed control instances using this Holonic Control solution, two process-driven or rule-driven Holonic Control instances over ANALYTICE II are here described. In the first instance, the production of parts X and Y was simulated in the system shown in Fig. (2). The Production Plan for X and Y were:

- X: {buffering on Store.1 [pos 1, 2, 3, 4, 5, or 6]}, {buffering on Table.1 [pos 1]}, {operation in Machine-Tool.1}, and {buffering Table.2 [pos 1 or 2]}.
- Y: {buffer on Store.1 [pos 7, 8, or 9]}, {buffering on Table.1 [pos 2]}, {buffering on Table.3 [pos 1]}, {operation in Lathe.1}, and {buffering on Table.3 [pos 2]}.

---

[1] Actually, the agent-notification is more than an elegant control solution. It represents an inference solution that eliminates searches and takes into account good practices of system engineering, such as functional independence between entities and avoidance of processing redundancies (Simão, 2005)(Simão et Stadzisz, 2008). This allows achieving suitable features in: (a) control, e.g. trade-off between determinism & reactivity, conflict identification & resolutions, and Petri net compatibility (Simão et al., 2003); (b) computation, e.g. excellent performance and openness to distribution (Simão et Stadzisz, 2002); (c) and systemic integration, e.g. rules are intuitive for humans and it is easy evolve the solution to be product-driven (Simão et al., 2006).

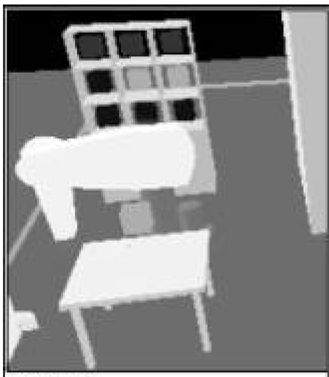
In agreement with the Production Plan, a Rule set was created allowing Resource-HL collaborations to carry out the production. For example, in Fig. (6) is presented a subset with Rules that allows carrying out the first production steps of the X and Y parts, i.e. these Rules allow transporting parts from Store.1 to the Table.1. The parts X are placed in position 1 to 6 of the Store.1 and after that they are transported to position 1 of the Table.1, whereas the parts Y are placed in position 7 to 9 of the same Store.1 and after that they are transported to position 2 of the same Table.1.

The complete set of Rules is presented in Simão et Stadzisz (2008) and Simão et al. (2008), as well as other experiment and set variations. In this sense, a third product type (Z) was introduced in this case study and the second type (Y) was enabled to use a second added lathe. This led to change Production Plan and rule sets, and then to observe that system adaptability for agility is feasible by changes of Rule-knowledge, as detailed in (Simão, 2005)(Simão et Stadzisz, 2008). However, it was also detected that Rules must be validated and simulation is, besides, an appropriate manner for this activity.

It was still simulated in that manufacturing-cell, presented in Fig. (2), the production of a real part by using another instance of the Holonic Control meta-model (i.e. the second instance). The part considered in the second instance is a real one produced by means of a lathe and a machining-tool, in the plant of AIPL (*Atelier Inter-établissements de Produtique – Lorraine*), a training place related to *Centre de Recherche en Automatique de Nancy* (CRAN) (Gouyon et al., 2004). Actually, the presented cell from Fig. (2) can be considered a simulation of AIPL cell except by the transport system.

In the simulated cell, the Production Plan to the real part is the following: {buffering on Table.3 [pos 1]}, {operation in Lathe.1}, {buffering on Table.3 [pos 2]}, {buffering on Table.1 [pos 1 or pos 2]}, {operation in Machine-Tool.1}, and {buffering on Table.2 [pos 1 or pos 2]}. Based on this Production Plan, a Rule set was also elaborated/used and statistical results were taken, as shown in Simão (2005). Briefly, the productivity was 83.68%, which is an acceptable production rate. Nevertheless, the Resource-HL loading/unloading time could be improved/optimised. Anyway, in the control viewpoint, the Rule Holons made their function of controlling Resource-HL collaborations.

Other experiments have also been made after an improvement of the Holonic Control solution as product-driven. These issues are discussed in the next section.



Rule A.1
If
Puma560.1 Status= Free
Table.1 Pos1 = Not_Busy
Store.1 Pos1 = Busy
Then
Puma560.1 Transport
(Store.1 Pos1, Table.1 Pos1)

Rule A.2
If Puma560.1 Status = Free
Table.1 Pos1 = Not_Busy
Store.1 Pos1 = Not_Busy
Store.1 Pos2 = Busy
Then Puma560.1 Transport
(Store.1 Pos2, Table.1 Pos1)

Rule A.3
If Puma560.1 Status= Free
Table.1 Pos1 = Not_ Busy
Store.1 Pos1 = Not_ Busy
Store.1 Pos2 = Not_ Busy
Store.1 Pos3 = Busy
Store.1 Pos7 = Not_ Busy
Then Puma560.1 Transport
(Store.1 Pos3, Table,1 Pos1)0

Rule A.4
If
Puma560.1 Status = Free
Table.1 Pos1 = Not_Busy
Store.1 Pos1 = Not_Busy
Store.1 Pos2 = Not_Busy
Store.1 Pos3 = Not_Busy
Store.1 Pos4 = Busy
Store.1 Pos7 = Not_Busy
Then
Puma560.1 Transport
(Store.1 Pos4, Table.1 Pos1)

Rule A.5
If
Puma560.1 Status= Free
Table.1 Pos1 = Not_ Busy
Store.1 Pos1 = Not_Busy
Store.1 Pos2 = Not_Busy
Store.1 Pos3 = Not_Busy
Store.1 Pos4 = Not_Busy
Store.1 Pos5 = Busy
Store.1 Pos7 = Not_Busy
Store.1 Pos8 = Not_Busy
Then
Puma560.1 Transport
(Store.1 Pos5, Table.1 Pos1)

Rule A.6
If
Puma560.1 Status = Free
Table.1 Pos1 = Not_Busy
Store.1 Pos1 = Not_Busy
Store.1 Pos2 = Not_Busy
Store.1 Pos3 = Not_Busy
Store.1 Pos4 = Not_Busy
Store.1 Pos5 = Not_Busy
Store.1 Pos6 = Busy
Store.1 Pos7 = Not_Busy
Store.1 Pos8 = Not_Busy
Then
Puma560.1 Transport
(Store.1 Pos6, Table.1 Pos1)

Rule A.7
If Puma560.1 Status= Free
Table.1 Pos2 = Not_ Busy
Store.1 Pos1 = Not_ Busy
Store.1 Pos2 = Not_ Busy
Store.1 Pos7 = Busy
Then
Puma560.1 Transport
(Store.1 Pos7, Table.1 Pos2)

Rule A.8
If
Puma560.1 Status= Free
Table.1 Pos2 = Not_Busy
Store.1 Pos1 = Not_Busy
Store.1 Pos2 = Not_Busy
Store.1 Pos3 = Not_Busy
Store.1 Pos4 = Not_Busy
Store.1 Pos7 = Not_Busy
Store.1 Pos8 = Busy
Then
Puma560.1 Transport
(Store.1 Pos8, Table.1 Pos2)

Rule A.9
If
Puma560.1 Status= Free
Table.1 Pos2 = Not_Busy
Store.1 Pos1 = Not_Busy
Store.1 Pos2 = Not_Busy
Store.1 Pos3 = Not_Busy
Store.1 Pos4 = Not_Busy
Store.1 Pos5 = Not_Busy
Store.1 Pos6 = Not_Busy
Store.1 Pos7 = Not_Busy
Store.1 Pos8 = Not_Busy
Store.1 Pos9 = Busy
Then
Puma560.1 Transport
(Store.1 Pos9, Table.1 Pos2)

Figure 6. A subset of Rule Holons from a first case study for transporting parts from the Store.1 to the Table.2

4. HOLONIC CONTROL SOLUTION – PRODUCT-DRIVEN CONTROL

In the case of product-driven, the improvement of the considered Holonic Control solution is "simple": Smart-Product-HLs allocate Rules according to its "needs", which technically is its Production Plan steps. In practical terms, an expert associates each Production Plan step to one or even more Rules. This expert can even be an artificial agent that compares Smart-Product-HL "needs" and Rules/Resource-HLs "skills" aiming to connect them, as detailed in Simão (2008).

Figure 7. UML diagram of the Holonic Control solution upgraded as a product-driven Holonic Control solution.

Anyway, in this improved Holonic Control solution, a Rule Holon execution also depends of its allocation by a Smart-Product-HL, as modelled in Fig. (7). Moreover, in some cases, the Smart-Product-HL can itselt provide some data to the Rules/Resource-HLs.

In short, the main advantage of this product-driven solution is a better Holonic Manufacturing System adaptability/agility by profiting Rule gains that, in this context, is organisation and optimisation of Holon collaborations (Simão et al., 2006). The Holonic Control product-driven was tested in three cases, which were detailed in the first author’s Ph.D. thesis (Simão, 2005) and succinctly described in Simão et al. (2006).

The first case study used the simulated-cell shown in the previous section. It was used the simulation of two (virtual) part-types, whereas the other simulation of a (real) part-type was not used once it does not use production flexibility. In this first case study, the observation was that products could be put in any place of the Store.1 and Rules are simpler, once Smart-Product-HLs allocate the correct Rules to arrive in Table.2 Position1 or Table.2 Position2 (depending of its type) and inform their position to them, as presented in Fig. (8) and detailed in Simão et al. (2008).

In short, the product-driven control allows an independence of position pre-allocated in the store, i.e. it could be put *n* (*n* < 10) products Y and *m* (*m* < 10-*n*) products X in the store, allowing a better buffer utilisation (Simão et al. 2006)(Simão et al., 2008). In the process-driven case, this would be complicated (at least) once each position was pre-allocated to a given part-type being this information used to compose and carry out the Rules.

Rule A.1	**Rule A.2**
If	**If**
Puma560.1 Status = Free	Puma560.1 Status = Free
Table2P.1 Pos1 = Not_Busy	Table2P.1 Pos2 = Not_Busy
Store3x3.1 Status = Not_Empty	Store3x3.1 Status = Not_Empty
Then	**Then**
Puma560.1 Transport (Store3x3.1. Pos?, Table2P.1. Pos1)	Puma560.1 Transport (Store3x3.1. Pos?, Table2P.1.Pos2)

Figure 8. Product-driven Rules – Rules that depends on a Smart-Product-HL allocation and even data.

The second case study was related to a Flexible Assembling Cell (FAC) from AIPL, a real Manufacturing System for engineering students presented in Fig. (9). FAC can assemble six pedagogical product types (A, B, C, D, E, and F shown in Fig. (9)) from six base part types (09, 01, 88, 11, 60, and 10) buffered in workstations (WSs). There are a WS for loading pallets that move on a conveyor, four WSs for assembling products on pallets stopped in face, and one WS for unloading products. Certainly, the type of base-part in WS for assembling defines the product type that it can assemble on the pallets. In turn, the conveyor has segments for pallet buffering and stopping-identification positions for deciding if a pallet have to visit a given WS or not. Each pallet can carry four products and has a digital memory for product information (Gouyon et al., 2004).

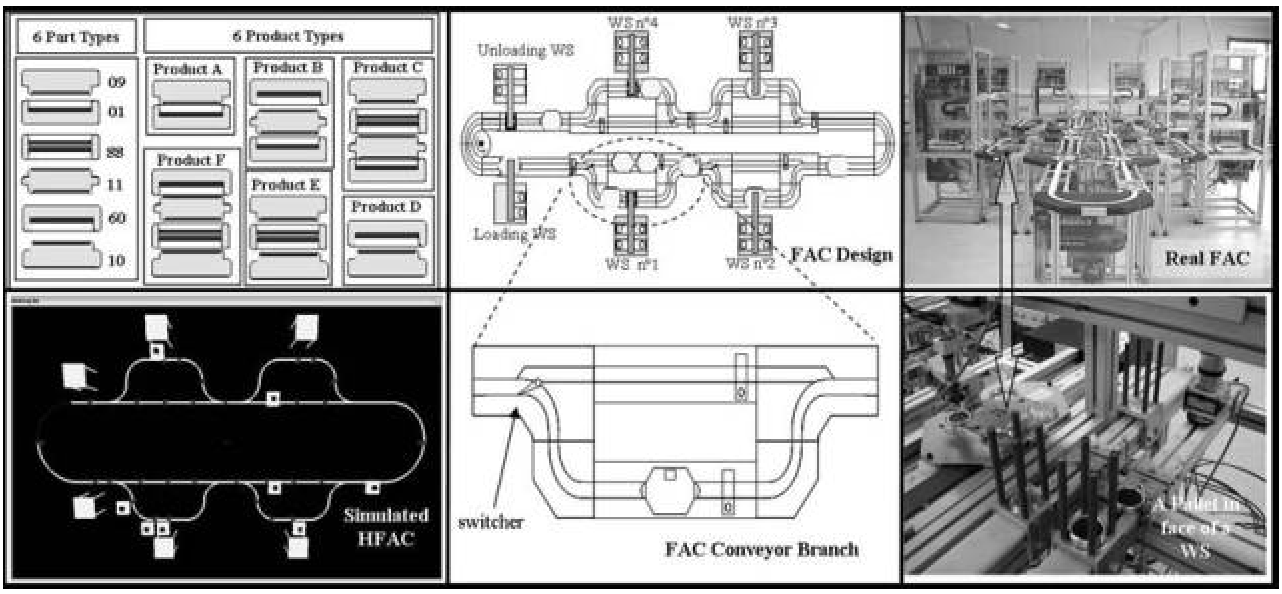


Figure 9. The FAC and products types made therein (Simão et Stadzisz, 2007).

In this second study case, by means of models and simulation efforts, it was observed that the product-driven Holonic Control solution allows exploiting the FAC flexibility, as detailed in Simão et Stadzisz (2007). Examples of that exploitation are: (a) Smart-Product-HL searching Rules to reach an alternative WS-HL when the aimed WS-HL is not available (e.g. very-loaded or even semi-loaded in the case of WS balance) and (b) Smart-Product-HL allocating Rules to go out of the system when production are not possible (e.g. some WS-HLs are in failure state) and avoiding the creation or launching other of its type in the system.

Actually, the presented solution of product-driven Holonic Control has been mainly applied to case studies related to manufacturing-cells, whose results have been subjects of the aforementioned publications. However, there is a case study related to a simulated plant using the solution, which is detailed in first author's thesis and was not yet detailed as paper. Therefore, as evolution of authors' works, this paper details that case study, which is related to a manufacturing-plant in a context of customised and just-in-time production. The plant is inspired by the AIPL shop floor dynamics, which is composed of manufacturing-cells and intermediated resources, drafted in Fig. (10). In this case study, the plant was holonified and simulated in ANALYTICE II by means of the proposed Holonic Control technology.

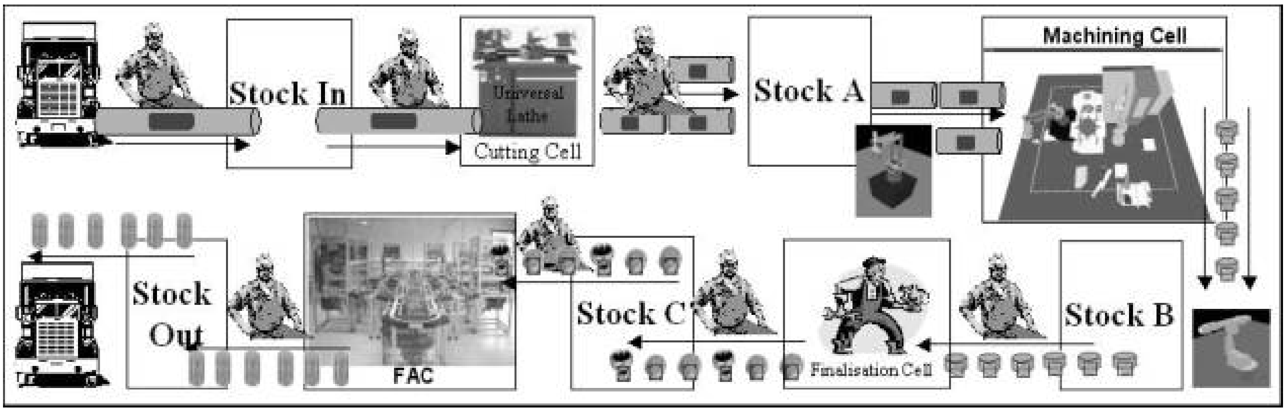


Figure 10. A similar production context to AIPL (*Atelier Inter-établissements de Produtique – Lorraine*).

5. CASE STUDY - HOLONIC PLANT

The third case study was about the realistic plant drafted in Fig. (10). In this plant, there is a FAC-HL for assembling six types of products (i.e. A, B, C, D, E, and F types) from base-parts stored in its WS-HLs, which can be of six types (09, 01, 88, 11, 60, and 10). There is also a Finalisation-Cell-HL that concludes and sends base-parts to the FAC-HL, a Machining-Cell-HL that produces and sends base-parts to the Finalisation-Cell-HL, and a Cutting-Cell-HL that cuts bars and sends the rough pieces to the Machining-Cell-HL.

Each production-cell and each intermediary resource was simulated as a black-box in ANALYTICE II emulation-part. Also, for each emulated-resource, an agent in the simulator control-part allowed composing a Resource-HL with suitable Attributes and Methods[2]. After that, it was elaborated a set of Rules based on the notification of Resource-HLs, as imposed by the Holonic Control solution. Actually, the Rules allow controlling the cooperation of those Resource-HLs.

Rule Produce-Product-Type-B	**Rule** Produce-Product-Type-E	**Rule** Product-Finished
If	**If**	**If**
FAC Part_Type_01_Inside = True	FAC Part_Type_60_Inside = True	FAC Product_Finished = True
FAC Part_Type_11_Inside = True	FAC Part_Type_08_Inside = True	Store-Out Position_Free = True
FAC Part_Type_10_Inside = True	FAC Part_Type_09_Inside = True	**Then**
Then	**Then**	FAC Liberate_Product_Finished.
FAC Start_Production_Product-Type-B	FAC Start_Production_Product-Type-E	Store-Out Receive_Product_(Type?).
FAC Decrement_Number_Part-Type-01	FAC Decrement_Number_Part-Type-60	
FAC Decrement_Number_Part-Type-11	FAC Decrement_Number_Part-Type-08	
FAC Decrement_Number_Part-Type-10	FAC Decrement_Number_Part-Type-09	

Figure 11 – Examples of product-driven Rule for FAC assemblages.

[2] The Resource-HLs related to production-cells could have a more refined simulation. In this case, their Attributes and Methods would be based on the Attributes and Methods of its internal Resource-HLs. For example, the FAC-HL has an Attribute called *Part_Type_01_Inside*, for determining if there is at least a base-part 01 inside, which would be based on Attributes of given WS-HLs related to base-part-01 number. Surely, the causal-relation of external and internal Attributes would be given by Rules. Anyway, the used simulation is enough to present the Holonic Control solution is a plant.

Some Rules are allocable by pre-established types of Smart-Product-HLs and other Rules are independent of them. The former are called product-driven Rules and the latter are called process-driven Rules. Actually, the Rule type depends upon its role in the system. Anyway, in general, the Rules are useful to Smart-Product-HLs be produced once they control the Resource-HL cooperation in the plant.

In the considered plant, the goal is to produce six different types of products without previsions (i.e. just-in-time). Thus, for each created production-order, a related Smart-Product-HL is created, which starts its own production in the FAC-HL by allocating a suitable Rule. For example, a Smart-Product-HL *Type B* would allocate the Rule (Holon) *Produce-Type-B* presented in the causal-rule form in Fig. (11). This allocation allows its production in the FAC-HL, in a given moment. After the production in the FAC-HL, each Smart-Product-HL uses the Rule *Product-Finished* to reach the Stock-Out-HL.

The production in the FAC-HL decreases the number of its base-parts. Thus, when the FAC-HL has fewer than an established limit of a given type of base-parts, it notifies a certain Rule, using the notification principle imposed by the Holonic Control meta-model. This notified Rule triggers the creation of an appropriate Smart-Product-HL related to the wanted base-part. For example, when the FAC-HL achieves a low number of base parts of type 09, the Rule *Start_Smart_Product_For_Part-Type-09* (presented in Fig. (12)) is notified aiming at its execution.

Rule Start_Smart_Product_For_Part-Type-09	**Rule** Start_Smart_Product_For_Part-Type-88
If	**If**
FAC Low_Number_Part_Type_09 = True	FAC Low_Number_Part_Type_88 = True
Then	**Then**
AIPL Start_Smart-Product-HL(Part-Type-09)	AIPL Start_Smart-Product-HL(Part-Type-88)
FAC Wait-For(Part-Type-09)	FAC Wait-For(Part-Type-88)

Figure 12 – Examples of process-driven Rules for launching base-part Smart-Product-HLs.

Each base-part Smart-Product-HL allocates some given Rules for production, thereby reaching the Machining-Cell-HL, following to the Finalisation-Cell-HL, and arriving in the WS-HLs. For example, a Smart-Product-HL related to a base-part-09 allocates the Rule *Production_Part_Type_09* (Fig. (13)) for achieving the Machining-Cell-HL. After its production in this cell, it still allocates the Rule *Store_Machined_Part* for achieving the StockB-HL.

Rule Production_Part-Type-09	**Rule** Store_Machined_Part
If	**If**
StockB Low_Number_Parts_09 = True	StockB Position_Free = True
StockA Cut_Bar_Inside = True	Mach-Cell Part_Finished = True
Mach-Cell Receive_Part_Possible = True	**Then**
Then	Mach-Cell Liberate-Part-Finished
StockA Liberate-Product.	StockB Receive_Product(Part-Type?)
Mach-Cell Receive_Product(Part-Type-09)	

Figure 13 – Examples of product-driven Rules related to the machining process.

Subsequently, that Smart-Product-HL allocates the Rule *Production_Part-Type-09* (Fig. (14)) for reaching the Finalisation-Cell-HL. After the finalisation, it allocates the Rule *Store_Finalised_Part* to reach the Stocks-HL. Finally, it allocates the Rule *Store_Part_09_FAC_WS2* finalising its own production.

Actually, the production of base-parts decreases the number of rough-base-parts in the Stock A. Thus, when the level of base-bars therein is fewer than an established limit, the Rule *Cut_Bar* (Fig. (15)) is notified aiming at the cutting of bars in the Cutting-Cell-HL. After the cutting of bars, the Rule *Store_Cut_Bar* is notified aiming at the transport of the cut bar (i.e. rough-parts) to the StockA-HL. These Rules are process-driven and not product-driven because there is not flexible production of rough-parts, i.e. they are the same for all base-parts.

Rule Finalisation_Part-Type-09	**Rule** Store_Finalised_Part	**Rule** Store_Part_09_FAC_WS2
If	**If**	**If**
StockC Low_Number_Parts_09 = True	StockC Position_Free = True	StockC Parts_09_Inside = True
StockB Machined_Parts_09_Inside = True	Finalis-Cell Part_Finished = True	FAC Parts_09_Wanted_WS2 = True
Finalis-Cell Receive_Part_Possible = True	**Then**	**Then**
Then	Finalis-Cell Liberate-Part-Finished	StockC Liberate_Part(Part-Type-09)
StockB Liberate- Part(Part-Type-09)	StockC Receive_Product(PartType?)	FAC Receive_ Part (WS2)
Finalis-Cell Receive_ Part(Part-Type-09)		

Figure 14 – Examples of product-driven Rules related to the finalisation process.

These two process-driven Rules represent the last production enchainment triggered when a product is ordered in the FAC-HL. This entire scenario of enchained production was simulated in ANALYTICE II by using a HMES based on the proposed Holonic Control meta-model, thereby demonstrating its applicability in a different production level than manufacturing-cell.

This HMES permitted to simulate the production without previsions, i.e. just-in-time, as observed on the graphical module of the simulator and verified by statistical data. For example, if a product *type F* is demanded, this results in four suitable base-parts produced to supply the changed stocks and also in the cutting of a bar to generate other four rough-parts.

Moreover, this experiment allowed the concomitant application of the Holonic Control both on process-driven and product-driven models. Examples of process-driven control Rules are those to decide good moments to create Smart-Product-HLs, whereas the examples of product-driven control Rule were those to allow Smart-Products-HL achieving their production goals.

Rule Cut_Bar	**Rule** Store_Cut_Bar
If	**If**
StockIn Rough_Bar_Inside = True	Cutting-Cell Cut_Bar_Inside = True
Cutting-Cell Receive_Bar_Possible = True	StockA Position_Free = True
StockA Position_Free = True	**Then**
Then	Mach-Cell Liberate_Cut_Bar
StockIn Liberate_Bar	StockA Receive_Cut_Bar
Mach-Cell Receive_Bar	

Figure 15 – Examples of process-driven Rules for launching bar production.

6. CONCLUSIONS

This paper presents a control approach for Holonic Manufacturing Execution System (HMES) of Holonic Manufacturing System. This solution firstly comprises the resource holonification, based on computational homogenisation, which generates Resource-HLs with expertise for expressing their states and receiving service requests. In fact, this expertise allows Resource-HLs carrying out control functions, namely monitoring and command. Subsequently, Rules are presented as organisers of Resource-HL collaborations, allowing process-driven HMES. The Rules are also Holons that work based on a previously proposed inference process: a notification-oriented method (Simão et Stadzisz, 2002)(Simão et Stadzisz, 2008).

The paper follows presenting a solution improvement to deal with Smart-Product-HLs, in which each Smart-Product-HL indirectly reserves Resource-HLs by allocating suitable Rules. Each Rule coordinates Resource-HL services to carry out a given Smart-Product-HL desire. Thus, a solution to product-driven HMES is given by these Holon collaborations regulated by Rules. In the product-driven context, Rules are a decoupling mechanism between Smart-Product-HL and Resource-HL collaborations, thereby allowing their organisation and optimisation.

This Holonic Control architecture has its first contribution in the ANALYTICE II holonification, facilitating composition of simulated Holonic Manufacturing Systems. In fact, Holonic Control experiments have been developed therein, demonstrating the solution generality. As result, it is considered a meta-model to Holonic Control (i.e. HMES), firstly in this simulation environment. Nevertheless, the solution can be also understood as an actual solution for real Holonic Manufacturing Systems once similar controlled Resource-HLs and Smart-Product-HLs have been developed in Intelligent Manufacturing System community, namely by McFarlane (2003).

On a mass-customisation point of view, this paper presents HMES tools to support and examine product-driven benefits. The results agree with those from Intelligent Manufacturing System literature, namely the capacity to produce parts from possible product-types without prevision. Moreover, an advantage of the presented approach is the organisation and the information enabled by Rules and their notification net. In this paper, these features were particularly observed by means of the proposed case study of a holonic plant, which is an additional meta-model application.

The case study of holonic plant has confirmed the approach properties and allowed to investigate new details, such as concomitant use of process-driven and product-driven Rules. Moreover, the case study has highlighted the actual use of the approach in larger context than manufacturing-cell. In fact, this highlighted that the Rules of the control solution works in the same manner over Resource-HLs with different kernels (such as machines, cells, plants and potentially other entities) whether the standard interaction interface (i.e. Attributes and Methods) is preserved.

Beyond the generality or fractal property of the control solution, the case study has also confirmed that exploitation of Manufacturing System flexibility, for agility and customization, depends on enabled information on holons, namely Rules, Resource-HLs, and Smart-Product-HLs. Finally, the case study has equally confirmed that ANALYTICE II is a suitable tool for realistic test of these holons and their relations.

The future works includes a deep development and evaluation of case studies about simulated and real applications. A new study example will be related to a real manufacturing-cell from UTFPR. It could be re-designed and simulated in ANALYTICE II by applying the Holonic Control meta-model. Also, the Holonic Control meta-model could be directly applied to the real cell, once it is decoupled from the simulator. Besides, taking into account the Resource-HL and Rule generality, other works about larger control-solution use are under development, highlighting application and advantages of the notification mechanism for inference and discrete systems in general (Banaszewski et al., 2007)(Simão et Stadzisz, 2008).

7. ACKNOWLEDGEMENTS

J. M. Simão would like to thank CAPES/Brazil (*Coordenação de Aperfeiçoamento de Pessoal Nível Superior*) for national/international Ph.D. scholarships between May 2001 and April 2004, as well as F. R. Banaszewski would like to thank CAPES/Brazil from his current M.Sc. scholarship. The authors would like to thank structure and resources offered by LSIP, CPGEI, CEFET-PR/UTFPR, UHP, CRAN, and AIPL. The authors would like to thank also for researcher collaboration at CPGEI/UTFPR and AIPL/CRAN/UHP. Finally, the authors would like to thank the ANALYTICE II authors (Koscianski et al., 1999), especially L. F. F. Rosinha (*in memoriam*) that additionally helped to develop the emulated FAC in the referred and related paper Simão et al. (2007).

8. REFERENCES

Banaszewski, R. F., Stadzisz, P. C., Tacla, C. A., and Simão, J. M, 2007, "Notification Oriented Paradigm (NOP): A Software Development Approach based on Artificial Intelligence Concepts". VI Congress of Logic Applied to the Technology – LAPTEC 2007, Santos, SP – Brazil

Da Silveira, G., Borenstein, D., & Fogliatto, F.S., 2001, "Mass customization: literature review and research directions." International Journal of Production Economics, vol. 72, pp 1-13,

Deen, S.M, 2003, "Agent-Based Manufacturing: Advances in the Holonic Approach". Springer. ISBN 3-540-44069-0.

Gouyon, D., Simão, J. M., Khaled, A, & Morel, G., 2004, "Product-driven issues for B2M-control systems integration". INCOM 2004 – 11th IFAC Symposium on Information Control Problems in Manufacturing, Salvador-BA Brazil.

Koscianski, A, Rosinha, L. F. F., Stadzisz, P. C., & Künzle, L. A, 1999, "FMS design and analysis: developing a simulation environment". Proceedings of the 15th Intern. Conf. on Cad/Cam, Robotics and Factories of the Future, Águas de Lindóia, Brazil, vol. 2.

McFarlane, D., Sarma, S., Chrin, J. L., & Ashton, K., 2003, "Auto ID systems and intelligent manufacturing control". Intelligent Manufacturing: Special issue of Eng. Applications of Artificial Intelligence, vol. 16, n. 4, pp. 365-376.

Morel, G. & Grabot, B., 2003, (Eds.). "Special issue of Engineering Applications of Artificial Intelligence". Intelligent Manufacturing, vol. 16, n.4.

Morel, G., Panetto, H., Zaremba, M. B., & Mayer, F, 2003, "Manufacturing Enterprise Control and Management System Engineering: Rationales and Open Issues", Annual Reviews in Control.

Qiu, R., Wysk, R., & Xu, Q, 2003, "Extended structured adaptive supervisory control model of shop floor controls for an e-manufacturing system". International Journal of Production Research, vol. 41, n.8, pp. 1605-1620, 2003.

Simão, J. M., 2005, "A Contribution to the Development of a HMS simulation tool and Proposition of a Meta-Model for Holonic Control". Ph. D. Thesis, Federal University in Technology of Paraná (CPGEI/UTFPR) & Henry Poincaré University (CRAN/UHP).

Simão, J. M. & Stadzisz P. C., 2002, "An Agent-Oriented Inference Engine applied for Supervisory Control of Automated Manufacturing Systems" Frontiers in Artificial Intelligence and Applications: Advances in Logic, Artificial Intelligence and Robotics LAPTEC 2002, vol.85, pp. 234-241, IOPRESS, Amsterdam, ISBNI 586032925.

Simão, J. M. & Stadzisz, P. C., 2007, "Manufacturing Execution System for Customised Production – Application Issues". 4º Congresso Brasileiro de Engenharia de Fabricação (COBEF), 2007 – São Pedro - SP.

Simão, J. M. & Stadzisz P. C., 2008, "Inference Based on Notifications: A Holonic Meta-Model Applied to Control Issues". IEEE Transactions on Systems Man & Cybernetics, Part A. Approved Paper waiting for publication. Reference number: SMCA07-07-0207.R1.

Simão, J. M., Stadzisz, P. C., & Morel, G, 2006, "Manufacturing execution system for customised production". Journal of Material Processing Technology, vol. 179, issues 1-3, Elsevier (from 3rd COBEF).

Simão, J. M., Stadzisz, P. C., & Künzle, L. A., 2003, "Rule and Agent-Oriented Architecture to Discrete Control Applied as Petri Net Players". Frontiers in Artificial Intelligence and Applications: Advances in Intelligent Systems and Robotics *LAPTEC 2003*. Vol. 101, pp. 121-129, IOS PRESS, Amsterdam - The Netherlands, ISBN I586033867.

Simão, J. M., Tacla, C. A., & Stadzisz, P. C., 2008, "Holonic Control Meta-Model". Transactions on Systems Man & Cybernetics, Part A. Paper submitted on July 7th 2007, evaluated as acceptable after a set of improvements in August, 2008, and under current author's improvement.

Valckenaers, P, 2001. *Special issue: Holonic Manufacturing Systems*. Comp. in Industry, vol. 46, issue 3, pp. 233-331.

Van Brussel, H., Wyns, J., Valckernaers, P., Bongaerts, L., & Peeters, P., 1998. Reference architecture for holonic manufacturing systems PROSA. *Computers in Industry*, vol. 37, pp. 255-274.

9. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.