



**HAL**  
open science

## Active product driven control of dynamic routing in FMS

Thierry Berger, Nadine Zbib, Yves Sallez, Damien Trentesaux

► **To cite this version:**

Thierry Berger, Nadine Zbib, Yves Sallez, Damien Trentesaux. Active product driven control of dynamic routing in FMS. *Journal Européen des Systèmes Automatisés (JESA)*, 2009, 43 (4-5), pp.407-434. hal-01876146

**HAL Id: hal-01876146**

**<https://hal.science/hal-01876146v1>**

Submitted on 18 Sep 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# Active product driven control of dynamic routing in FMS

**Thierry Berger, Nadine Zbib, Yves Sallez, Damien Trentesaux**

LAMIH-CNRS

Le mont Houy

Université de Valenciennes et du Hainaut-Cambrésis

59313 Valenciennes cedex 9

{name.surname}@univ-valenciennes.fr

---

*ABSTRACT. This paper deals with the concept of active product and its application to the control of flexible manufacturing systems. After presenting our major assumptions, we introduce the key element of decisional entities and its application to the definition of the notions of active product and active resource. The proposed control structure is then presented. Dynamic allocation algorithms for such active products and/or resources are shortly presented, and a highly distributed stigmergic approach to routing control is then detailed. The routing control model is based upon a functional architecture with two levels: a virtual level in which virtual active products evolve stochastically in accelerated time, and a physical level in which physical active products, with embedded decisional capabilities, evolve deterministically in real time. We illustrate the model's adaptive capabilities through several simulation scenarii using NetLogo software, and present an on-going real implementation (where "intelligence" is embedded within each product) taking place at the AIP-PRIMECA Center in Valenciennes, France..*

*RÉSUMÉ. Cet article traite du concept de « produit actif » et de son application au pilotage des systèmes de production flexible. Après avoir introduit nos principales hypothèses nous présentons la notion d'entités décisionnelles et l'application de ce concept pour définir les notions de produit actif et de ressource active. La structure de pilotage est alors proposée. Un algorithme d'allocation dynamique de ces produits et ressources est présenté, et une approche stigmergique totalement distribuée pour le routage est ensuite détaillée. Ce modèle de routage repose sur une architecture fonctionnelle à deux niveaux : un niveau virtuel où les produits actifs virtuels évoluent de manière stochastique en temps accéléré, et un niveau physique où les produits actifs physiques, avec capacité décisionnelle réellement embarquée, évoluent de façon déterministe en temps réel. Les capacités d'adaptation du modèle sont illustrées à l'aide de plusieurs scenarii en utilisant la plate-forme de simulation NetLogo, et nous présentons une mise en oeuvre en cours de réalisation (où chaque produit embarque son « intelligence ») au centre AIP PRIMECA de Valenciennes.*

*MOTS-CLÉS : stigmergie, routage dynamique, produit augmenté, pilotage, système de production flexible*

2 Revue. Volume X – n° x/année

*KEYWORDS: stigmergy, dynamic routing, augmented product, manufacturing control, flexible manufacturing system*

---

## 1. Introduction

This paper addresses the issue of production process control in flexible manufacturing systems (FMS), in which heterarchical relations exist between some decisional entities. One expected advantage concerns the fact that, since it is not possible to control or anticipate everything in real time at the upper levels, local adjustments within global constraints will help to support unexpected events or complexity by solving the problem as closer as possible at the location where the solution will be applied. For example, it is well known that calculating all the routings for all products is very complex, while letting the products negotiate with the transportation system to move from one section to another is easy to implement. The “chaos technology” proposed and industrialized by Montech (2008) uses this approach and we think that it is one another step towards a full product driven control of a production system..

First we introduce the global assumptions on which our proposal relies. Then we present the principle of heterarchy and decisional entities. Based upon these assumptions and a short analysis of the state-of-the-art, we introduce the concept of “augmented product”. The proposed control structure is then presented. In our approach, it is composed of two sub-systems, hierarchically dependent, one for the task allocation process, and the second, for the routing process given the allocation decisions. Since the task allocation control has already been studied (Sallez et al., 2004), we focus here on our innovative approach to routing and its implementation phase in a real FMS.

## 2. Global assumptions

To situate our study in the overall domain of FMS, it is very important to introduce the main assumptions on which our study is based and on which our model relies.

1. the topology of the FMS transportation system is assumed to be associated to a strongly connected, directed graph, in which nodes can be both production resources and disjunction points in the transportation system, and arcs model the areas of the transportation system that require no decisions during the routing process since products can only move in one direction towards the next node. Routing times are assumed to be non-negligible compared to production times and non necessary constant. Production resources are also assumed to be flexible, meaning that the same operation type can be executed on several resources, with the possibility of different processing times.

2. Products and production resources are assumed to be decisional entities. Usually, these products and resources are considered to be non decisional & passive entities: they never communicate, decide nor act during production process. However,

current technology (e.g., RFID, smart cards, embedded systems, wifi, and infrared communication) has recently led to some new activities involving “active” decisional entities, in which products and resources are able to act in accordance with the real state of the production system. This new ability can be embedded in the product or production resource itself (McFarlane et al., 2002). In this paper, products and production resources that can be taken as decisional entities are denoted AP (augmented products) or AR (augmented resources).

3. The studied production control system must control the production process in real time while being as optimized as possible. Then, to enable long term optimization, we assume that global constraints are provided from an upper level, e.g., from an ERP (enterprise resource planning) system. These constraints (inputs of our control system) are assumed to be information about planned manufacturing orders optimized over the long term (what to do and when?). The controlled production process is assumed to be composed of several activities, involving mainly the execution of manufacturing orders using production resources and the routing of products within the production system. Given this assumption, dynamic production control systems must deal with many issues: dynamic scheduling/task allocation (who undertakes the tasks required for each manufacturing order?), tool and inventory management, reconfiguration/preparation of resources, transportation system management, performance indicator updating, and so on. From an industrial information system point of view, such a research activity holds at the Manufacturing Execution level of production and may be concerned with so-called MES (Manufacturing Execution Systems). (Morel et al., 2007) have outlined some challenging issues for manufacturing execution: nonlinear nature of the underlying production system, uncertainties stemming from the production processes and the environment, combinatorial growth of the decision space. As a consequence, they argue that schedules and plans, originating from higher levels in a manufacturing organization, can become ineffectual within minutes on a factory floor since manufacturing is a very dynamic environment. As a consequence, instead of designing a complex centralized & hierarchical MES & ERP that will hardly cope with these dynamics and complexity by deciding all, our assumption is to relieve the MES & ERP levels of some decisional processes that should be managed in a easier way in a heterarchical control system while assuming that global constraints limiting the myopic behaviour of such a heterarchical control system are provided (mainly, planned manufacturing orders optimized over the long term). Concerned decisional processes are characterized by strong limiting assumptions. One of the processes is the routing one, often modelled as just a simple constant lag in an infinite capacity transportation system, which is, from our point of view very prejudicial to an efficient, accurate and reactive control of production flows. This paper focuses on this decision process.

4. The structure of the production control system is assumed to be non-centralized (i.e., no central data management system can be identified) and to have a non fully hierarchical control structure (this concept will be described more precisely later).

### **3. State-of-the-art and objectives**

This section focuses on the state-of-the-art of the literature in the domain of production control systems that are not fully hierarchical. Since a study of relevant contributions has been realized in Trentesaux (2007), we only provide the reader with the conclusion of this study. It has been shown that most research developments focus on distributed or decentralised dynamic control of one of the issues mentioned above: dynamic task allocation/scheduling. See, for example, studies by Aissani et al. (2008), Bousbia et al. (2005), Cavalieri et al. (2000), Lastra and Colombo (2006), Maione and Naso (2003), Tharumarajah (2001), Wong et al.(2006), Duffie and Prabhu (1996) and Baker (1998). Our initial work in this domain led us to also propose a dynamic allocation process based upon an heterarchical product-driven and resource-driven control of dynamic scheduling (Sallez et al., 2004). However, in order to apply concepts relevant to this domain in a real FMS, dynamic scheduling is not sufficient; several other process controls must also be developed (e.g., tool management, inventory level control, routing process control).

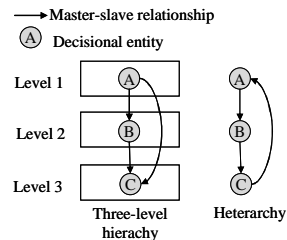
We are currently working on producing a semi-heterarchical control for a real FMS (AIP-PRIMECA Valenciennes Cell). This is why we must then now integrate all the previously introduced process controls into our model. Since, from the product point of view, the product alternatively searches for services among the available production resources (transformation) and the available transportation resources, we choose to begin by integrating routing process control with our previously designed allocation process control. This new model for routing process control, and its integration with our previous allocation model, is described in this paper. The next section defines precisely what we mean by heterarchy, which is important since it is a key concept for the designing of the new control structure.

### **4. Heterarchy**

In our opinion, heterarchy can be formalized by using graph theory. A directed graph composed of nodes representing decisional entities and arcs representing the master-slave interaction of a decisional entity (master) with another entity (slave) is called "influence graph". A decisional entity is a system able to support a decision process, that is, a process composed of a triggering activity (e.g., a distance between a desired goal and a evaluated state from sensors), a problem design and solving activity and the application of subsequent decisions (actuators).When each node can be considered both a master and a slave, no hierarchy can be identified. The graph is

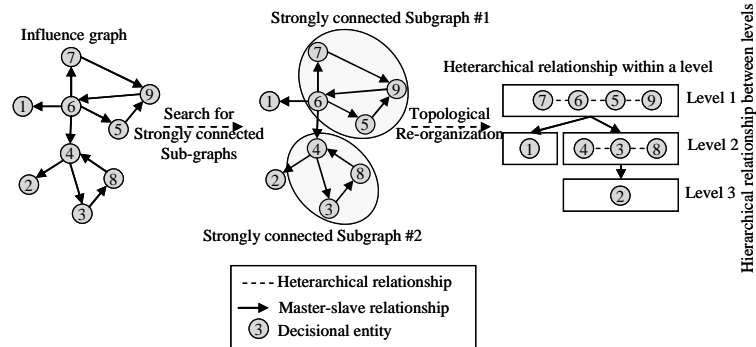
thus considered to be strongly connected, which defines a heterarchy. This formalisation is consistent with the initial heterarchy concept developed by (McCulloch, 1945). Figure 1 illustrates the difference between hierarchy and heterarchy graphically.

Thus, hierarchy and heterarchy are both relevant to the structure of a system composed of several decisional entities (cf. figure 2). Based on these two concepts, different structures can be formed, ranging from fully hierarchical (no heterarchical relationship) to fully heterarchical (no hierarchical relationship). We use the term "semi-heterarchical" to refer to a structure that can not be assimilated to a fully heterarchical system while it contains at least one sub-graph that is fully heterarchical.



**Figure 1.** *Hierarchy and heterarchy*

Applied to the context of this paper, a product or a production resource will be considered as a decisional entity. But a major difference arises when we consider the fact that, in production phase, resources are fully operational, while products are work in progress: products are not fully operational during the production and evolve from the beginning to the end of production (this is not the case when we consider the product in the use phase, as a resource can be in production systems). This transient state makes it difficult to consider it as a fully operational decisional entity during all the production process. As a consequence, past research developments have mainly focused on resources as the only decisional entities, see for example Duffie and Prabhu (1996), Baker (1998), Trentesaux et al. (1998). But nowadays, technological offer (embedded systems, communication...) make it also possible to integrate decision and communication capabilities into work-in-progress products.



**Figure 2.** Hierarchy and heterarchy as building blocks of a system composed of decisional entities

To integrate (embedded or distant) decisional and communication capabilities into WIP products to transform them into decisional entities, a possibility consists in using the transportation system and especially pallets to support these capabilities: an “augmentation module” would then be attached to products, the whole being an “augmented product” AP. This concept of augmentation would also be extended to define “augmented resource” AR. An augmented resource is a resource (e.g., machine, robot) with an assistant (hardware and/or software) that transform it into a decisional entity. In the literature, such entities are often called intelligent or activesince the concept of intelligent/active product is more recent, the following part focuses only on relevant litterature.

## 5. Active/intelligent product: a short review

In the manufacturing system, Bajic (2004) uses the term of a *communicating object* which is able to communicate with its environment, its users, and other objects or entities, generally those that can communicate and interact with the original object through wired or wireless communication technologies. Cea Ramirez (2006) and Dobre (2008), two of the people that collaborated with Bajic, have expanded this concept.

The term *active product* is used by Brun Picard (Brun Picard et al., 1997) in the context of manufacturing system control, as an autonomous entity cooperate and coordinate their actions to reach their goals.

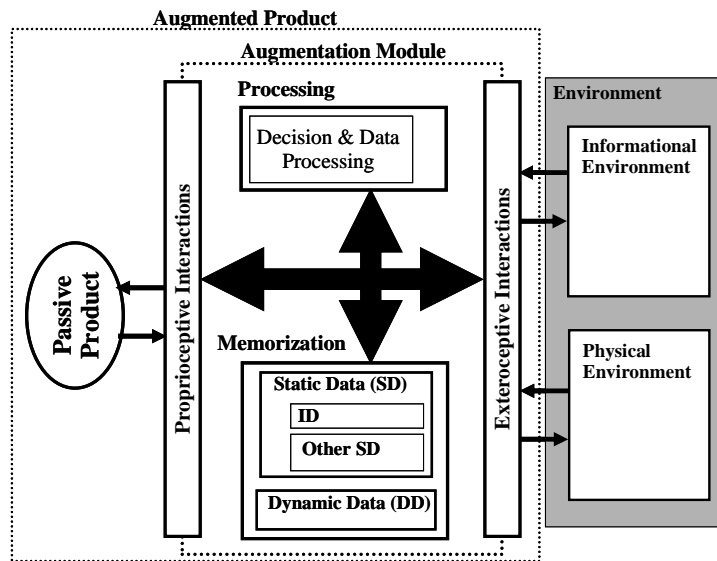
McFarlane et al. (2002) used the term, intelligent product, to refer to a communicating object in the manufacturing system, associating a set of capabilities with a commercial product that is equipped with an Auto ID system and advanced software. For them, the ability to communicate with the environment and to influence the decisions affecting its future was a product asset. These references are well described in (Zbib et al., 2008).



To summarize, we note that the meaning of the terms “active product” or “intelligent product” can differ greatly from one author to another. In addition, the proposed models & typologies are very technology-oriented and lack genericity. Based on our review, we propose a conceptual framework that remedies these problems, and we introduce the concept of augmentation module to obtain our augmented product. This model is designed to be relevant for the entire life cycle of intelligent products, though in the rest of the paper we focus specifically on the manufacturing phase.

## 6. Concept of Augmented product – Augmentation module

An *augmented product* is a passive product with an augmentation module will be able to transform it into decisional entity able to make appropriate decisions about its own destiny during its life cycle and able to interact with their environment. The augmentation modules of products can be removed before delivering to customers and be re-used in the production system for future products. Figure 3 shows a schematic version of an augmented product incorporating our augmentation module.



**Figure 3.** Schematic view of augmentation module for transforming a passive product into augmented product

The *augmentation module* has three components: Memorization, Processing and Interaction.

- *Memorization*: This component deals with both static and dynamic data.
- *Processing*: This component deals with basic “information-oriented” or more sophisticated “decision-making” data processing.

- *Interaction*: This component deals with interaction. Two kinds of interaction can be identified, either between the augmentation module and the product, or between other entities (i.e., products and resources) in the informational and/or physical environment. The first type is called *proprioceptive* interaction and the second is called *exteroceptive*. This type may involve *physical* interaction or *informational* interaction.

The next part describes our control structure integrating our augmented product concept.

## 7. Proposed augmented-product based control structure

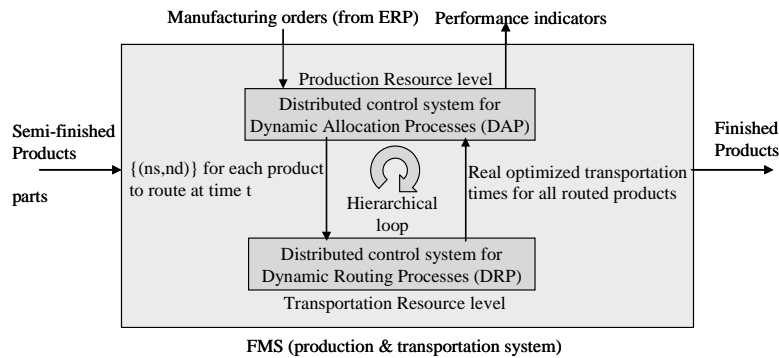
An augmented-product based control system is a production control system where products are decisional entities able to interact with each other (and possibly, with other resources considered also as decisional entities, e.g. augmented production resources) in order to reach assigned global goals. One expected advantage concerns the fact that, since it is not possible to control or anticipate everything in real time at the upper levels, local adjustments within global constraints will help to support unexpected events or complexity by solving the problem as closer as possible at the location where the solution will be applied. For example, it is well known that calculating all the routings for all products is very complex, while letting the products negotiate with the transportation system to move from one section to another is easy to implement. The “chaos technology” proposed and industrialized by Montech (2007) uses this approach and we think that it is one another step towards a full product driven control of a production system.

The proposed augmented-product control system is then designed to determine in real time the list of each successive task (operation, date) that an augmented resource AR has to realize, taking into account both allocation and routing issues. Thus, the two processes that must be controlled are the dynamic allocation process (DAP) and the dynamic routing process (DRP).

The proposed control structure, which is semi-heterarchical, is shown in Figure 4. In this structure, allocation is controlled prior to routing. Inputs to the DRP control system are assumed to be composed of the set of couples (ns, nd) – where ns is the augmented resource source node and nd the augmented resource destination node – that concern one or more products at a given time t. This information is defined by the DAP control system. Outputs from the DRP control system are the optimized real transportation times for routed products. This information will be used by the DAP control system to improve its allocations. Relationship between DAP & DRP control systems is then hierarchical.

Meanwhile, the global structure is partially heterarchical since each of the two control systems is fully heterarchical: no hierarchy can be identified in any of the individual control subsystems (decisional entities) and there is neither centralized

data management nor centralized processing capability. A fully heterarchical structure in which the DRP and DAP cooperate to optimize allocation and routing jointly is possible, but such a structure generates new issues that are currently being studied and have yet not been published.



**Figure 4.** *Proposed semi-heterarchical control structure*

The following provides the control algorithm supported by our previous control structure model.

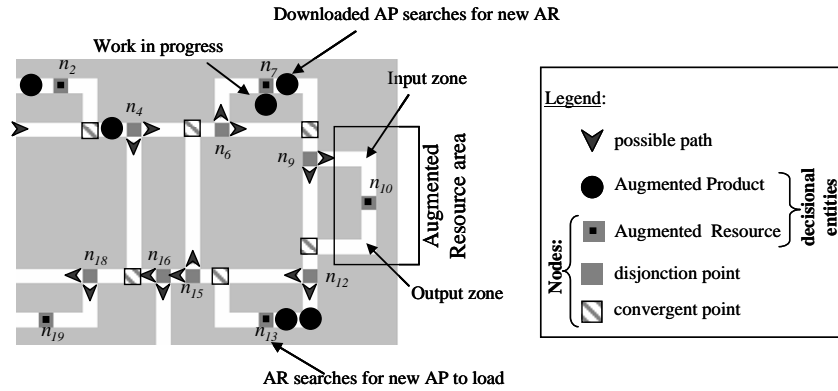
## 8. Proposed control algorithm

Our initial research dealt with a model of a theoretical heterarchical DAP control where augmentation concept was only concerning resources. We have shown that advantages of this basic control model are expressed in terms of reactivity and adaptability facing unpredictable events while providing effective scheduling performances in terms of completion times compared to statically optimized schedulings with no perturbation (Trentesaux et al., 2000, Trentesaux et al., 2001). We have, in a second step, also applied the augmentation concept to products (Sallez et al., 2004). Now, we improve on this control model, by combining it with the DRP control in a more global control algorithm, which allows the hierarchical articulation of the two control systems. This new approach is fully reactive since it assumes that manufacturing orders have been planned and optimized using a wider time horizon. Since our goal is to validate our model on real embedded systems, a distributed real-time programming approach was used (e.g., reduction of the number of loops and non-blocking states).

The following are the main notations used in our model.

Let  $N = \{n_i\}$  be the set of considered nodes  $n_i$  (augmented resources – AR) or disjunction points in the transportation system) and  $X = \{x_j\}$  be the set of augmented products – AP. Both AR and AP are decisional entities. For each AR, an input zone exists where an AP waits for a task (i.e., operation to be executed) and an output

zone where an AP waits to be transferred. Figure 5 illustrates these notations and shows the two decisional entities AR and AP.



**Figure 5.** Notations and decisional activities.

The main algorithm (below) features the two kinds of decisions made by AR and AP:

When an operation is completed by AR  $n_i$  on AP  $x_j$   
 AP  $x_j$  is downloaded  
 AR  $n_i$  decisional algorithm is triggered  
 AP  $x_j$  decisional algorithm is triggered

These two decisional algorithms are described in the next two sections.

### 8.1. Augmented Resource decisional algorithm

When an AR has finished an operation on AP  $x_j$ , it downloads  $x_j$  to its output zone if space is available (pre-conditions). Then the AR analyzes its input zone, if several AP are available, it chooses the “best” next  $x_k$ . The algorithm responsible for this decision has two steps:

1. AR  $n_i$  identifies next  $x_k = g(D)$
2. AR  $n_i$  loads AP  $x_k$

In this algorithm, the function  $g$  depends on evaluating a criterion  $D$  that is defined according to the global objective assigned to the control system. The choice for the best criterion to achieve this objective is still an open issue. However,  $D$  is usually based upon such information as shortest processing time, shortest remaining time/tasks or first-in first-out to name a few. The best product to load is the one that

proposes the best value for D through mono-criterion (min or max) or multi-criteria analysis.

Steps 1 & 2 are supported by the DAP heterarchical control system. Historically, our design choices for this DAP level inherits from previous studies where all decisions were attached to resources, defining the so-called IMS (integrated management station) (Trentesaux et al., 1998). We still attach such decisions to production resources mainly because of the fact that resources can be assigned with specific performance indicators and objectives such as workload management, utilisation rate, etc. which are non-sense for products. Product loading decisions can be taken by augmented products, but, as a consequence, with a lack of visibility for the managers. More globally, this discussion generates a very interesting issue: what are the advantages/drawbacks of attaching decision capabilities only to resources, or only to products, or to both of them. A first incomplete study has been led (Bousbia et al., 2005) and must be improved.

### **8.2. Augmented Product decisional algorithm**

When an operation is completed on an AP, the AP enters the output zone of the AR that has executed the operation. It must then choose (decide) an AR for the next operation (service). To do so, the AP uses the following 3-step algorithm:

1. AP  $x_j$  identifies the next operation to execute
2. AP  $x_j$  reserves the “best” destination AR  $n_d = f(D', M)$
3. AP  $x_j$  moves from current node AR  $n_s$  to AR  $n_d$

The  $f$  function depends on two parameters: the criterion  $D'$  used to optimize the operation and the method  $M$  used to gather information. The best choice for criterion  $D'$  is, like the choice for  $D$ , open to debate, but this criterion is usually based upon such information as next available time, shortest remaining number of operations to proceed and/or transfer time, for example. In a heterarchical context, one of the most usual methods chosen for  $M$  is based upon the contract-Net protocol (Smith, 1980), but a blackboard system could also be used.

Steps 1 & 2 are supported by the DAP control system. In our current development, step 2 applies the contract-net to define a negotiation protocol dedicated to dynamic task allocation. This negotiation protocol helps to identify the “best” AR destination for each requesting AP, as follows: when an operation in a particular manufacturing order has been performed, the subsequent AP  $x_j$  sends a **request** through the communication network about the next operation to be processed (according to the routing list of the AP's manufacturing order). Each of the AR  $n_k$  able to perform this operation returns an **acceptance** and enough local information to allow  $x_j$  to evaluate the response (in our case, the estimated next

available time, given the size of the product queue and the work in progress). The requesting AP  $x_j$  then selects one of the proposed AR by minimizing the  $D'$  criterion (i.e., the next available time, given the current real travel time to each AR candidate provided by the DRP system). The requesting AP  $x_j$  sends a **reservation** to the selected destination AR  $n_d$  and a **release** to the others  $n_{k \neq d}$ . A **discharge** sent to  $x_j$  from the selected AR  $n_d$  concludes the protocol and validates the allocation “contract”. Then,  $n_d$  is given, AP  $x_j$  being at AR  $n_s$ . Step 3 can thus be realized at the DRP level.

### 8.3. Looping DAP and DRP control systems

The input for the DRP control system is then the set of couples  $(n_s, n_d)$ . After the routing operation has been executed,  $\tau_{nd}^*(n_s)$ , denoting the minimal real-time to go from AR  $n_s$  to the destination AR  $n_d$ , is then updated by the DRP control system. This value is sent back to the DAP control system each time the  $D'$  criteria must be estimated. As a result, the real state of the transportation system can be taken into account when determining the “best” AR  $n_d$ . The hierarchical loop between the two control systems is then pointed out.

The following section underlines the innovative aspect of our approach and describes in detail the DRP control system, its simulation and its validation on a real FMS cell. The validation of the coupling of the two control systems will be presented in another paper.

## 9. The DRP control system

### 9.1. Concept of stigmergy

To make the DRP adaptive and robust, our approach is based on the concept of stigmergy. The term “stigmergy” describes the mechanism used during ants foraging activities (Theraulaz and Bonabeau, 1999). Ants find food and carry it back to the nest, simultaneously laying down a pheromone trail. Other ants, detecting the pheromones, follow the trail back to the food source. As more ants bring food to the nest, they each reinforce the chemical trail on the path they follow. Since pheromones tend to evaporate over time, the more attractive trails accumulate more pheromones and thus have an advantage over the other trails. Over time, due to the natural reinforcement of the ants, only the shortest trail remains.

Several researchers (Dorigo and Stützle, 2006; Parunak et al., 2001; Brückner, 2000; Peeters et al., 1999; and Hadeli et al., 2004) have applied the stigmergy concept to solve optimization problems or specific situations in manufacturing control systems. In most stigmergic applications (e.g., ants searching for food), time is not really considered and attempts cost the user nothing; only the final solution is taken into

account. However, in a production context, time inefficiency is prejudicial to the system: for example, if too many solutions are tried before finding the right one (e.g., too many shuttles are used to try paths), the overall performance quality rapidly deteriorates, for instance, in terms of completion time. Consequently, a model is needed that will identify satisfactory solutions quickly in order to accelerate the convergence of algorithms as much as possible. In response to this need, we propose a functional two-level—physical and virtual—architecture for the heterarchical control of DRP. A close approach was used by (Qiu, 2005) via the introduction of virtual production lines in a semiconductors manufacturing context.

## **9.2. Modelling approach**

Figure 6 shows the proposed architecture with its two levels and the intermediary Data Space (DS) that memorizes the information for both levels:

- The Virtual Level (VL), where Virtual Augmented Products (VAPs) move, is an informational domain in which everything is simulated in accelerated time as quickly as possible (and reduce considerably the learning phase and adaptation of our stigmergic algorithm);
- The Physical Level (PL) represents the real world, in which Physical Augmented Products (PAPs) move in real-time.

A node  $n_i$  (disjunction point or AR) has three components: a virtual image (in the VL), a data memorization and processing structure (in the DS), and a physical infrastructure (in the PL). Since VL works in accelerated time, adaptation time becomes physically short. On the VL, lots of VAPs can be used to discover new paths, which help to keep the number of PAPs circulating on the PL level low. The virtual entities (VAPs) make decisions based on stigmergic principles, which include stochastic decision-making and allow adaptive behavior to deal with unexpected perturbations. VAP travel history is used to update the pheromones (i.e., routing data). From a logical point of view, VAP move within a model of the existing routing network in the PL, which contains PAPs. At the PL, PAP routing decisions are made deterministically with “best efforts”, based upon the optimized results of the VL.

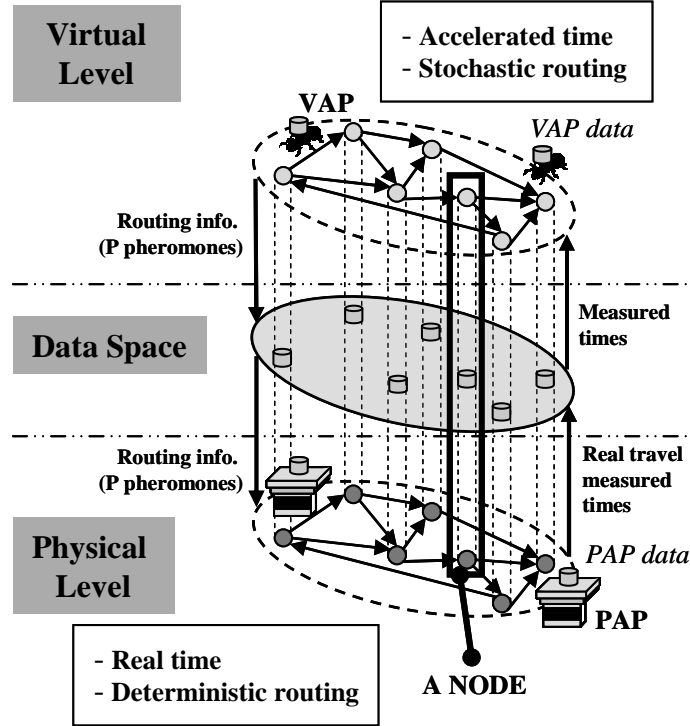


Figure 6. The two levels architecture

The following section introduces the notations used in this model, which is then described more precisely.

### 9.3. Notations and variables

The variable  $v_{n_i}^w$  denotes the  $w^{\text{th}}$  neighbour of  $n_i$  and  $V_{n_i}$ , the set of the neighbors of  $n_i$ :  $V_{n_i} = \{v_{n_i}^w / \exists \text{arc } n_i \rightarrow v_{n_i}^w\}$ ,  $w \in \{1 \dots \text{card}(V_{n_i})\}$ .  $N = \{n_i\}$  and  $V_{n_i}$  thus describe then the topology of the FMS, including the transportation system. A possible path between node  $n_i$  and node  $n_j$  is written  $u_{ij} = [n_i \dots n_j]$  and corresponds to the ranked list of successive nodes to be visited in order to travel from  $n_i$  to  $n_j$ .

Variables:

- $t_{n_i}(v_{n_i}^w)$  and  $t_{n_i}^*(v_{n_i}^w)$  are, respectively, the travel time (measured by a PAP) and the reference time (for a VAP) between a node  $n_i$  and its  $w^{\text{th}}$  neighbor.



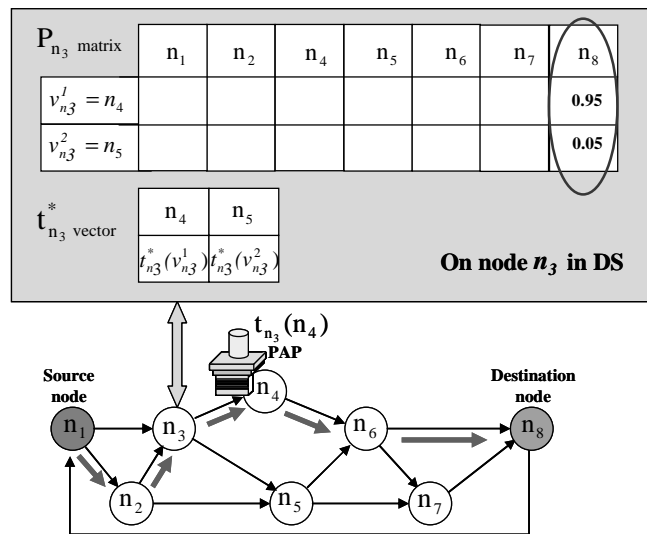
–  $P_{n_i}$  is the pheromone matrix. Columns of  $P_{n_i}$  correspond to all possible destination nodes  $n_j$  from  $n_i$  and rows contain all existing neighbor nodes  $v_{n_i}^w$ . In analogy to biological systems, this matrix characterizes the pheromone rate on the different arcs  $n_i \rightarrow v_{n_i}^w$ . The value  $P_{n_i}(v_{n_i}^w, n_j)$  is associated to the preferred path from the current node  $n_i$  to destination node  $n_j$ , when choosing the possible intermediate neighbor node  $v_{n_i}^w$ . Consequently, the following expression is true:

$$\sum_w P_{n_i}(v_{n_i}^w, n_j) = 1, \forall (n_i, n_j) \in N^2$$

In this paper, the best path is considered to be the path on which the PAPs spend the least time.

Figure 7 gives an example of the use of the above variables. Let a PAP be set on current node  $n_i = n_3$ , coming from the source node  $n_1$  with the destination node  $n_j = n_8$ . In this example, the best neighbor is  $n_4$  since  $P_{n_3}(v_{n_3}^1 = n_4, n_8) = .95 > P_{n_3}(v_{n_3}^2 = n_5, n_8) = .05$ . PAP will then travel from  $n_3$  to  $n_4$ , keeping its destination node  $n_8$  in mind. When arriving at  $n_4$ , the time spent  $t_{n_3}(n_4)$  is reported to node  $n_3$  via node  $n_4$  and is used by the node  $n_3$  to determine whether or not a perturbation (e.g., bottleneck, slow down, break down) has occurred between  $n_3$  and  $n_4$  by comparing  $t_{n_3}(n_4)$  with  $t_{n_3}^*(n_4)$ .

–  $\tau_{n_i}^*$  is the minimum time vector.  $\tau_{n_i}^*(n_j)$  thus denotes the minimum measured time to move from node  $n_i$  to any possible destination node  $n_j$ . This variable is updated every time a lower time is clocked, according to a mobile time window (not described in this paper).



**Figure 7. Variables illustration**

–  $h_{vap}(n_i, n_j)$  is the history table. Each VAP traveling between nodes  $n_i$  and  $n_j$  updates its history table  $h_{vap}(n_i, n_j)$ , which is used to store the nodes reached and the corresponding arrival dates:

$$h_{vap}(n_i, n_j) = ([n_i, n_k, n_m, \dots, n_u, n_j], [d_i, d_k, d_m, \dots, d_u, d_j])$$

where  $d_i$  is the arrival date at node  $n_i$ .

Based on the previous notations, the time spent to travel from  $n_i$  to  $n_j$  is

$$\tau_{n_i}(n_j) = d_j - d_i.$$

**9.4. Model of the routing process control**

9.4.1 Nominal routing process control

At a given time, a PAP begins to travel from its AR source node  $ns$  to its AR destination node  $nd$ . The pair  $(ns, nd)$  is provided by the DAP control system. The current node is designated  $nc$ , thus  $nc = ns$  when the PAP begins to move. The values of  $P_{nc}(v_{nc}^w, nd)$  are used to choose the next neighbor node  $mn$ . This choice is deterministic since the neighbor with the greater P value is always chosen  $mn = \tilde{v}_{nc}^{nd}$  where:

$$\tilde{v}_{nc}^{nd} \in V_{nc} / P(\tilde{v}_{nc}^{nd}, nd) = \max_{w \in \{1, \dots, \text{card}(V_{nc})\}} P_{nc}(v_{nc}^w, nd)$$

During the real move on the arc  $nc \rightarrow mn$ , the real elapsed time  $t_{nc}(mn)$  is measured. When the PAP arrives at node  $mn$ , it moves backwards  $t_{nc}(mn)$  to node  $nc$ . A local perturbation is detected on the arc  $nc \rightarrow mn$  if  $|t_{nc}(mn) - t_{nc}^*(mn)| > \varepsilon$ , where  $\varepsilon$  is a fixed threshold parameter.

9.4.2. VAP exploration

When a perturbation is detected, a VAP exploration is triggered. At each time interval T, a VAP is generated on node  $ns$  and is assigned the  $nd$  destination node. On a current node  $nc$ , the VAP must choose the next neighbor node  $mn$ . Since both luck and diversity are important adaptation mechanisms in natural biological systems, the choice of  $mn$  is not deterministic (as is true for PAP) but stochastic. The

different  $P_{nc}(v_{nc}^w, nd)$  are then assigned the probability to reach, from nc and to the destination nj, the neighbour  $v_{nc}^w$ . However, if  $P_{nc}(v_{nc}^w, nd)$  is negligible,  $v_{nc}^w$  will rarely be chosen. To ensure diversity, a minimum value  $\alpha$  is introduced, making the relation:

$$\text{Prob}(nn = v_{nc}^a) = \max\left(\frac{P_{nc}(v_{nc}^a, n_j)}{\sum_w P_{nc}(v_{nc}^w, n_j)}, \alpha\right) = \max(P_{nc}(v_{nc}^a, n_j), \alpha) \text{ since } \sum_w P_{nc}(v_{nc}^w, n_j) = 1$$

which corresponds to the probability of node  $v_{nc}^a$  being chosen as the next neighbor (denoted nn).

After reaching node nn, the passage date  $d_{nn}$  is then stored in the history table  $h_{vap}$ . Because the move is virtual, this date is calculated by adding the time reference  $t_{nc}^*(nn)$  to the last date in history table.

Then the node nn becomes the new current node nc, and the VAP continues to travel to the destination node nd, iteratively.

Finally, when the VAP reaches the destination node, the routing is finished and the history table is reported to the node nd.

#### 9.4.3. The search for the “best” alternative path

Like biological ants, which lay down a pheromone trail when returning to the nest, every time a VAP reaches its destination node nd, the information contained in its history table  $h_{vap}$  is forwarded to each visited node to update pheromone matrices. In our approach, this coefficient updating process is based upon a reinforcement law.

When the forwarded information reaches node nc,  $\tau_{nc}(nd)$  and the minimal time  $\tau_{nc}^*(nd)$  are compared. If  $\tau_{nc}(nd)$  is less or equal to the minimal time  $\tau_{nc}^*(nd)$  and if nn was the neighbor node used to travel from nc to nd,  $P_{nc}(nn, nd)$  is reinforced:

$$P_{nc}(nn, nd) \Leftarrow P_{nc}(nn, nd) + r * (1 - P_{nc}(nn, nd)) \text{ with } nn \in V_{nc}$$

Coefficients  $P_{nc}(no, nd)$  for the destination nd of the other neighbors no are negatively reinforced through a process of normalization:

$$P_{nc}(no, nd) \Leftarrow P_{nc}(no, nd) - r * P_{nc}(no, nd) \text{ with } no \in V_{nc}, no \neq nn$$

The “best” path emerges when all the  $P_{nc}$  coefficients at the reached nodes nc are superior to a fixed threshold  $\Omega$  (0.95 in our study). As shown in figure 11a, this state is reached after a transient period (Zone A). The path is declared “best” after a verification/check period (Zone B), which is indispensable to be sure that the found

path is good enough. This last period is finished when a fixed number of VAPs have reached the destination node  $nd$ .

## 10. Simulation of the DRP control system

### 10.1. *Short description of the simulation tool*

The proposed control model is naturally distributed, meaning there is no central memorization and processing system. This property influenced our choice of an agent-based parallel modeling and simulation environment. With NetLogo (Wilensky, 1999), each modeled entity can be described as an independent agent interacting with its environment. All agents operate in parallel on a grid of patches (i.e., a cellular world), and each agent can read and modify some of the attributes linked to the patches in its proximity. The behavioral rules defined for the agents make it possible to describe agent-environment interaction, which is very important when simulating the stigmergic process. Netlogo was chosen because it provides an intuitive, and well-documented programming language with an elegant graphical interface. NetLogo comes with extensive documentation, including a library with over 150 sample models and is very easy to learn. It's a very efficient tool for fast prototyping of multi-agents systems. In the following case study, both the physical and virtual levels are simulated. Figure 8 shows our simulator's interface. An applet is available at url: <http://www.univ-valenciennes.fr/sp/routing/>.

On figure 8, "SETUP" and "GO" buttons are respectively used to load the network and to run the simulation. Several switches "perturb" are available near arcs to generate a flow reduction on the corresponding arc. The different monitor windows show the chosen destination node, the simulated time evolution or indicate the different simulation phases (VAP exploration, Check period, PAP moves). The plots are relative to the pheromone rate evolution on nodes  $n_4$ ,  $n_6$  and  $n_9$ .

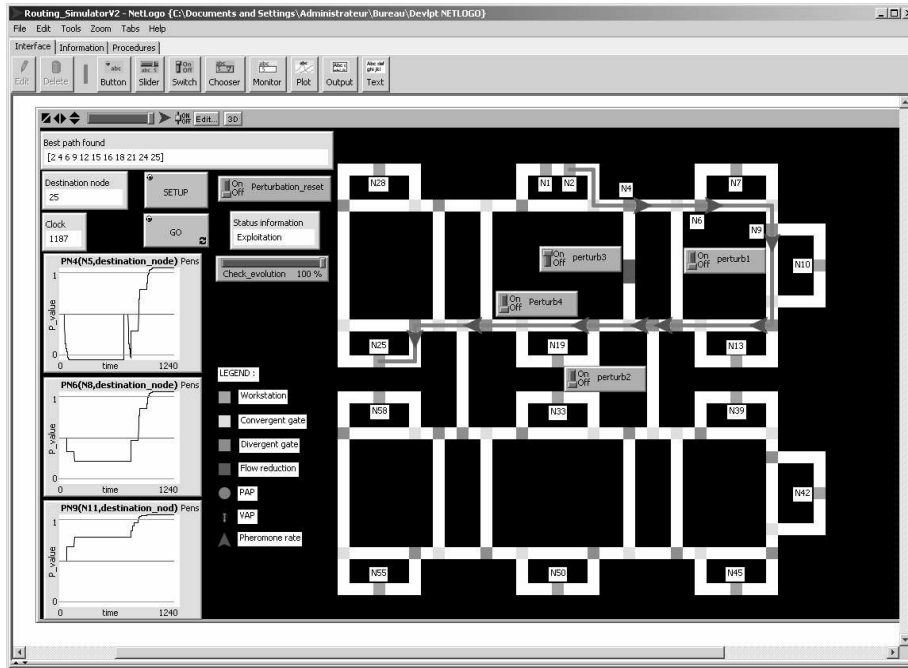


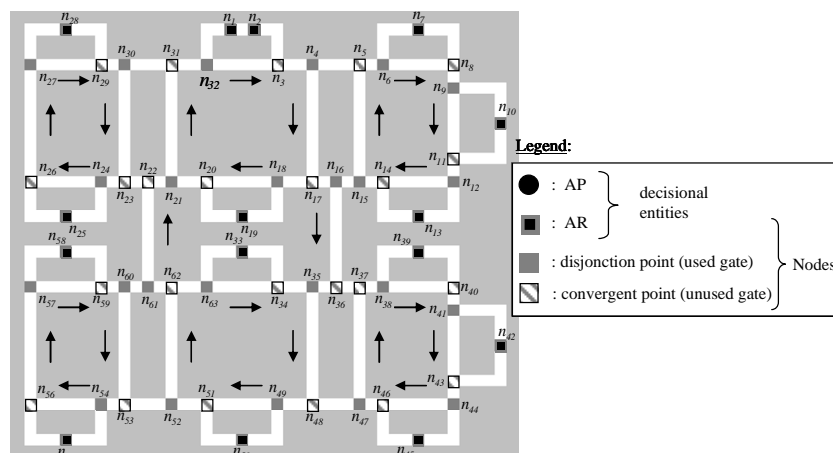
Figure 8. Simulator interface

### 10.2. Case study

The proposed approach is currently applied to an FMS built around a conveyor network based on a one-way conveyor system with divergent transfer gates that allow PAPs to be routed toward the different workstations. This FMS, shown in figure 9, can be characterized as follows:

- 15 workstations (nodes in medium grey with a black square inside),
- 24 divergent transfer gates (nodes in medium grey), where a routing choice between two adjacent arc must be made,
- 24 convergent transfer gates (nodes in grey with cross bars).

The FMS is divided into two parts – upper and lower – connected by two arcs  $n_{16} \rightarrow n_{36}$  and  $n_{61} \rightarrow n_{22}$ . The upper part is a replica of the flexible assembly cell of the Valenciennes AIP-PRIMECA center, which is used as experimental support (figures 14, 15).



**Figure 9.** *FMS conveying/resource network***10.3. Results**

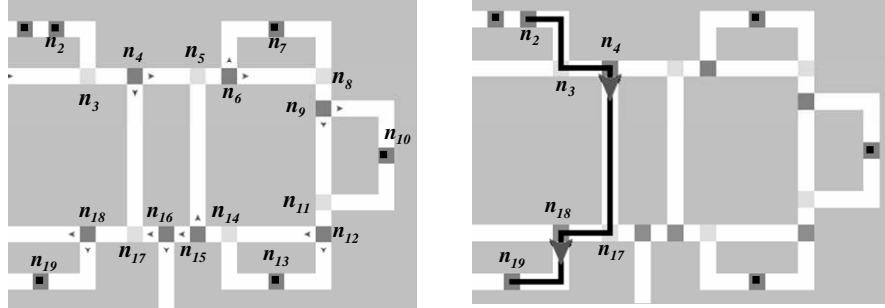
Several scenarii are presented, pointing out the different capabilities of our model with respect to self-adaptation to perturbations. These scenarii are summarized in table 1.

– **Scenario 1:** In the first scenario, the impact of a perturbation on an arc that can be by-passed is studied.  $n_2$  and  $n_{19}$  are the source and destination nodes, respectively. Qualitative & quantitative results are presented.

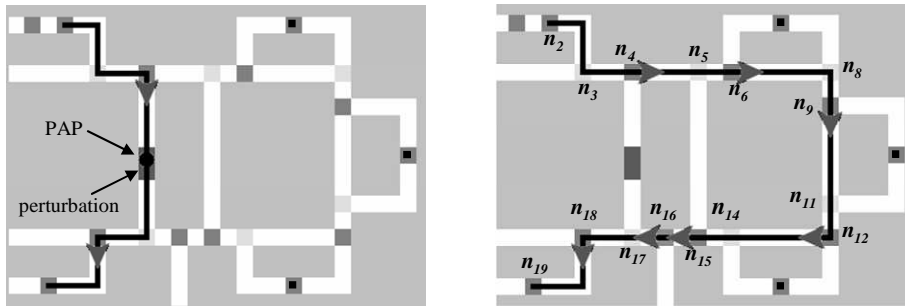
**Qualitative results:** Figure 10a shows the beginning of the initialization phase. Arrows (near divergent transfer gates) represent the pheromone rate corresponding to the adjacent arcs. Their size is proportional to the pheromone rate (initial values set to 0.5). The VAPs travel randomly along the different arcs, and the path  $[n_2, n_3, n_4, n_{17}, n_{18}, n_{19}]$  emerges through reinforcement. At each divergent transfer gate, the bigger arrows indicate the better arcs. The best path is then traced (in black), cf. figure 10b.

Figure 10c indicates the presence of a perturbation on the arc  $n_4 \rightarrow n_{17}$ , affecting traffic fluidity on this arc. (The flow reduction is shown in dark grey). When a PAP (black disc) takes a longer time on this trajectory than expected, a new VAP exploration is triggered via  $n_4$ .

The VAPs travelling on the arc  $n_4 \rightarrow n_{17}$  perform poorly, thus the appeal of this path decreases. The decreased appeal of this perturbed path consequently increases the appeal of an alternative path  $[n_2, n_3, n_4, n_5, n_6, n_8, n_9, n_{11}, n_{12}, n_{14}, n_{15}, n_{16}, n_{17}, n_{18}, n_{19}]$  (figure 10d), even though this path was originally not the best. This dynamic response is a classic display of the natural routing reconfiguration capacity of the stigmergic approach.



Figures 10a, 10b. Initial state; “best” path found



Figures 10c, 10d. Perturbation triggered; alternative path

**Quantitative results:** Figure 11 shows the evolution of some of the  $P_{n_i}$  coefficients. Please note the change that occurs after the perturbation at date  $d=350$  (in simulated time). Figure 12 shows more precisely the evolution of the coefficient  $P_{n_4}(n_{18}, n_{19})$  for a perturbation occurring at  $d=902$  and for its resolution at  $d=1472$ . After the perturbation has been resolved, the coefficient  $P_{n_4}(n_{18}, n_{19})$  again converges to 1.

– **Scenario 2:** In the second scenario, a perturbation is introduced on arc  $n_{16} \rightarrow n_{36}$ . This arc is an obligatory bridge to travel from the upper to the lower section of the FMS and it can't be by-passed. Please note that, after the perturbation, that the alternative path found is the same as the initial path (cf. table 1). This is possible because the minimum reference time  $\tau_{n_{10}}^*(n_{39})$  was constructed using a mobile time window, built with the 20 last measured times. In this context, the previous “good” performance of  $\tau_{n_{10}}^*(n_{39})$  is forgotten and the new longer time becomes the

reference time. This scenario outlines the good adaptability of our approach in a variable environment.

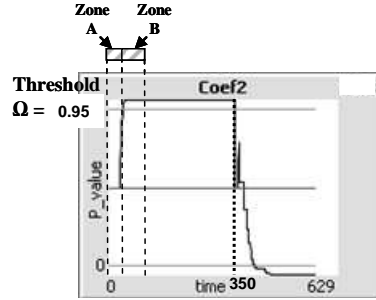


Figure 11a.  $P_{n_4}(n_{18}, n_{19})$

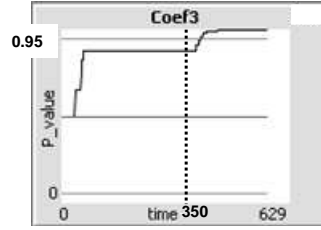


Figure 11b.  $P_{n_6}(n_9, n_{19})$

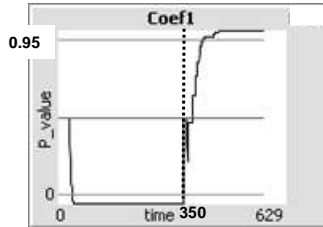


Figure 11c.  $P_{n_4}(n_6, n_{19})$ .

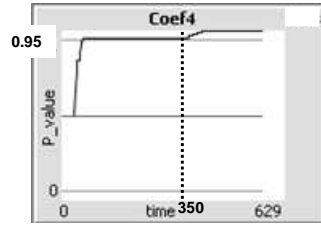


Figure 11d.  $P_{n_9}(n_{12}, n_{19})$

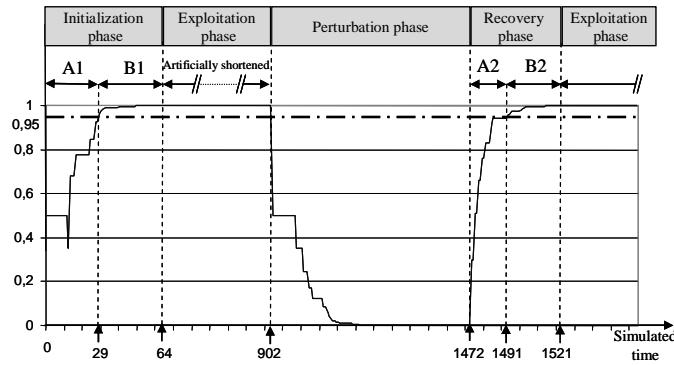


Figure 12. Evolution of  $P_{n_4}(n_{18}, n_{19})$

– **Scenario 3:** Nodes  $n_2$  and  $n_{25}$  are the source and destination nodes, respectively.

In the event of a perturbation on the arc  $n_{20} \rightarrow n_{21}$ , the initial simple path becomes:

$[n_2, n_3, n_4, n_5, n_6, n_8, n_9, n_{11}, n_{12}, n_{14}, n_{15}, n_{16}, n_{36}, n_{37},$

$n_{38}, n_{40}, n_{41}, n_{43}, n_{44}, n_{46}, n_{47}, n_{48}, n_{49}, n_{51}, n_{53}, n_{54}, n_{56}, n_{57}, n_{59}, n_{60}, n_{61}, n_{22}, n_{23}, n_{24}, n_{25}]$ .



This scenario outlines the capability of our stigmergic approach to find an interesting alternative path, very different of the initial path.

Source-Destination nodes	Initial path	Perturbed arc	Alternative path identified
$n_2$ to $n_{19}$ (Scenario 1)	$n_2, n_3, n_4, n_{17},$ $n_{18}, n_{19}$	$n_4 \rightarrow n_{17}$	$n_2, n_3, n_4, n_5, n_6, n_8, n_9, n_{11},$ $n_{12}, n_{14}, n_{15}, n_{16}, n_{17}, n_{18}, n_{19}$
$n_{10}$ to $n_{39}$ (Scenario 2)	$n_{10}, n_{11}, n_{12}, n_{14},$ $n_{15}, n_{16}, n_{36}, n_{37},$ $n_{38}, n_{39}$	$n_{16} \rightarrow n_{36}$	$n_{10}, n_{11}, n_{12}, n_{14}, n_{15}, n_{16}, n_{36},$ $n_{37}, n_{38}, n_{39}$
$n_2$ to $n_{25}$ (Scenario 3)	$n_2, n_3, n_4,$ $n_{17}, n_{18}, n_{20},$ $n_{21}, n_{22}, n_{23},$ $n_{24}, n_{25}$	$n_{20} \rightarrow n_{21}$	$n_2, n_3, n_4, n_5, n_6, n_8, n_9, n_{11}, n_{12},$ $n_{14}, n_{15}, n_{16}, n_{36}, n_{37}, n_{38}, n_{40},$ $n_{41}, n_{43}, n_{44}, n_{46}, n_{47}, n_{48}, n_{49},$ $n_{51}, n_{52}, n_{53}, n_{54}, n_{56}, n_{57}, n_{59},$ $n_{60}, n_{61}, n_{22}, n_{23}, n_{24}, n_{25}$

**Table 1.** Scenarii description

## 11. Elements for a real implementation

Given the promising results of the simulation, we decided to test our model on a real implementation, respecting the assumptions introduced in section 1.

Three types of equipment were used:

- *A Conveyor system:* Based on Montrac technology (Montech, 2007), this system is a monorail transport system using self-propelled shuttles to transport materials on tracks;
- *Node instrumentation, (figure 13):*
  - a “gate controller” which works to oversee the transfer gate and to help avoid collisions;
  - a DS containing  $P_{nc}, t_{nc}$ , the two variables required by the PL;
  - two data communication systems (*Ethernet* for node-to-node interaction, and *IrDA* for node-to-shuttle communication (esb-101 Clarinet system (Clarinet System, 2007)); and
  - a data processing system, supported by a Wago 750-841 controller (Wago, 2007), which is shared by the VL and PL.
- *Shuttle instrumentation:* This instrumentation is based on a data processing system embedded in a PDA device as augmentation module (including an IrDA communication) (figure 14).

Infra-red technology was chosen because of its low energy consumption for a high transfer rate (more than 100 Kb/sec), its short range, which enables geographical node localization, and its light-based physical layer, which is naturally robust to electromagnetic noise.

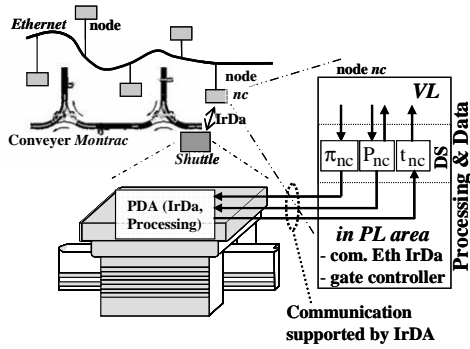


Figure 13. Schematic representation

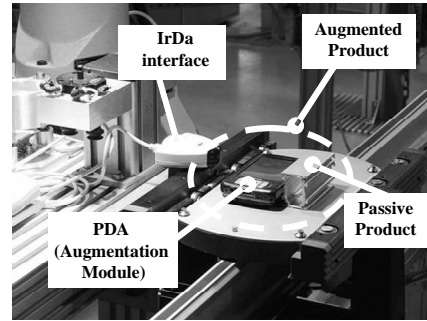


Figure 14. Real implementation

The figure 15 describes in details the localization of the future equipments within the flexible assembly cell of the Valenciennes AIP-PRIMECA center. The communications between node and product were successfully validated, and Wago programs are now under development.

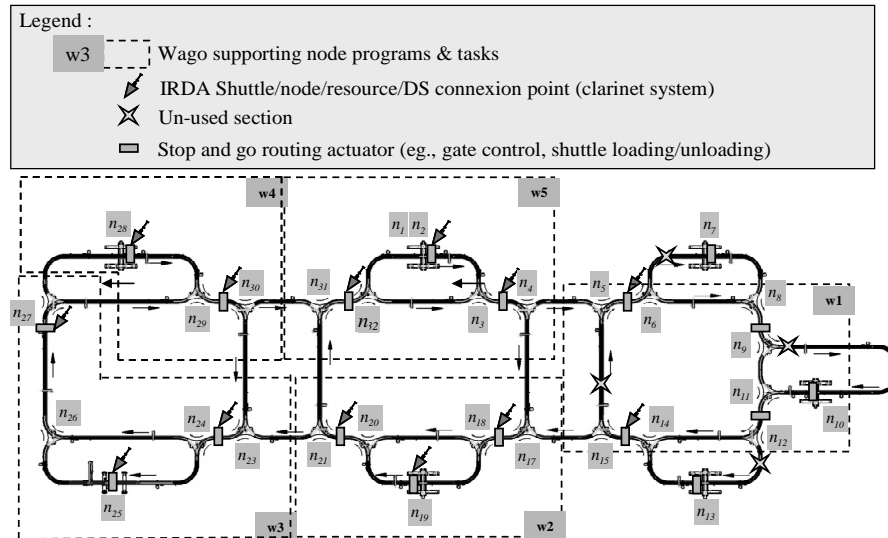


Figure 15. AIP cell case study

## 12. Conclusion and prospects

The results described above illustrate the advantages of our approach. First, the PAPs are able to determine the best path from the departure node to the destination

node without any centralized control. Second, they are able to surmount perturbations by seeking out new paths that bypass the perturbation but still lead to the desired destination. Other simulations under studies but not described in this paper highlight the fact that the alternative found path converges towards the optimal path. The main advantage of our approach is that none of the assumptions on which the system is based relies on the usual simplifications, such as unlimited stock capacity, no possibility for jamming, unlimited reliability of the transportation system, and so on. Our proposal should lead to more realistic and deployable real-time control systems.

Two interesting perspectives can be identified:

– First of all, to extend the results of our study, an holonic approach for AP modelling is under study: an AP is then considered as being recursively composed of other APs or of terminal passive products. A passive product, without an augmentation module, is composed only of raw material elements. A major application area of this holonic point of view is assembly manufacturing, where many augmented products may interact. With available ambient intelligence technologies, the insertion of an incompatible part in a sub-assembly can be prevented, because the interactions between the two augmentation modules will detect the incompatibility.

– Our second main perspective concerns the interaction of intelligent products via our augmentation module. Such interaction must be studied in more depth in order to design an efficient & effective global manufacturing control. A classic approach to control is to use the contract-net protocol (Smith, 1980) to integrate negotiation mechanisms in the AP, but potential functionalities of AP in ambient intelligence context are not exploited. We are currently working on a more innovative idea called “open control”, encompassing the classic explicit control measures with a new kind of control that is implicit. Contrarily to “explicit control” where global goals are translated in local objectives within distributed entities, the “implicit control” assumes that the entities are influenced by their environment (e.g. with stigmergic approach) and by other entities in an heterarchical way. Still, the two kinds of control (explicit and implicit) must be integrated within a larger control paradigm. Consequently, different ways to link our DRP model with the DAP distributed control system must be investigated.

### 13. Bibliography

- Aissani N., Trentesaux D., Beldjilali B., Use of Machine Learning for Continuous Improvement of the Real Time Manufacturing Control System Performances, International Journal of Industrial and Systems Engineering (to appear, 2008).
- Bajic E, “Ambient Networking for intelligent objects management, mobility and services”. Seminar Institut For Manufacturing - IFM University of Cambridge , Angleterre, 2 April 2004.
- Baker A.D, A Survey of Factory Control Algorithms That Can Be Implemented in a

- Multi-Agent Heterarchy: Dispatching, Scheduling, and Pull, *Journal of Manufacturing Systems* 17(4), 1998, p. 297-320.
- Baker A.D, « A Survey of Factory Control Algorithms That Can Be Implemented in a Multi-Agent Heterarchy: Dispatching, Scheduling, and Pull », *Journal of Manufacturing Systems*, Vol. 17, n° 4, 1998, p. 297-320.
- Bousbia S., Anli A., Trentesaux D., Grislin E., Agile scheduling of flexible manufacturing systems of production, 17th World Congress – IMACS 05, CD-Rom, (Paris, Juillet 2005), paper # T6-R-00-0316, 8p.
- Brückner S., Return from the Ant synthetic ecosystems for manufacturing control, Thesis Humboldt-University of Berlin, June 2000.
- Brun Picard D, Bouvet H, Baboli H and Binder Z, “The product as an active element of distributed production control”. IFAC-IFIP-IMACS Conference, Belfort, 20-22 May 1997.
- Cavaliere S., Garetti M., Macchi M., Taisch M., An experimental benchmarking of two multi-agent architectures for production scheduling and control, *Computers in Industry*, vol. 43, 2000, p. 139-152.
- Cea Ramirez A, Thesis “Contribution à la Modélisation et à la Gestion des Interactions Produit-Processus dans la Chaîne Logistique par l’Approche Produits Communicants”, Research center of Automatic in Nancy, 18 July 2006.
- Clarinet System, Network Connectivity for Mobile Devices, <http://www.clarinetsys.com>, 2007.
- Dobre D and Bajic E, “Active product modeling for chemical security management based on smart object concept”. 7th Francophone Conference of MODélisation and SIMulation - MOSIM’08, Paris – France.
- Dorigo M., Stützle T., *Ant Colony optimization*, The MIT Press, 2004.
- Duffie N.A., Prabhu V., Heterarchical control of highly distributed manufacturing systems, *international Journal of Computer Integrated Manufacturing*, vol. 9, n°4, 1996, p. 270-281.
- Hadeli T., Valckenaers P., Kollingbaum M. & Van Brussel H., Multi-agent coordination and control using stigmergy, *Computers in industry*, vol. 53, pp. 2004, p.75-96.
- Lastra J.L.M, Colombo A.W., Engineering framework for agent-based manufacturing control, *Engineering Applications of Artificial Intelligence*, vol. 19, 2006, p. 625–640.
- McCulloch W. S., Heterarchy of values Determined by the Topology of Nervous Nets, *Bull. math. biophys.*, vol. 7, 1945, p. 89-93.
- McFarlane D., Sarma S., Chirn J.L., Ashton K., The intelligent product in manufacturing control and management. 15th triennial IFAC world congress, Barcelone, Spain, 2002.
- Montech Technology, Conveyor systems, <http://www.montech.com>, 2007
- Maione G., Naso D., A soft computing approach for task contracting in multi-agent manufacturing control, *Computers in Industry*, vol. 52, 2003, p. 199–219.

- Morel G., Valckenaers P., Faure J. M., Pereira C. E., Diedrich C., Manufacturing plant control challenges and issues, *Control Engineering Practice* 15 (2007) 1321–1331.
- Parunak H.V.D., Brueckner S., Sauter J., ERIM's Approach to Fine-Grained Agents, *Proceedings of the NASA/JPL Workshop on Radical Agent Concepts (WRAC'2001)*, Greenbelt, MD, Sept. 19-21, 2001
- Peeters P., Van Brussel H., Valckenaers P., Wyns J., Bongaerts L., Heikkilä T. & Kollingbaum M., Pheromone based emergent shop floor control system for flexible flow shops, *Proc. of International Workshop IWES'99*, Kobe, Japan, Dec. 6-7, 1999.
- Qiu R.G., Virtual Production Line Based WIP Control for Semiconductor Manufacturing Systems, *International Journal of Production Economics* 95(2), 2005, pp. 165-178.
- Sallez Y., Trentesaux D., Berger T., Tahon C., Product-based and resource-based heterarchical approaches for dynamic FMS scheduling, *Computer and Industrial Engineering*, vol. 46, 2004, p. 611–623.
- Smith R. G., The contract net protocol: High level communication and control in a distributed problem solver, *IEEE Transactions on Computer*, vol. C29, n°12, 1980, p. 1104-1113.
- Trentesaux D., Les systèmes de pilotage hétérarchiques : innovations réelles ou modèles stériles ?, *Journal Européen des Systèmes Automatisés*, Lavoisier, ISSN 1269-6935, vol. 41, n°9-10, 2007, pp. 1165-1202.
- Tharumarajah A., Survey of resource allocation methods for distributed manufacturing systems, *Production, Planning & Control* 12(1), 2001, p. 58-68.
- Theraulaz G. & Bonabeau E., A brief history of stigmergy. *Journal of Artificial Life*, vol. 5, No 2, 1999, p. 97-116.
- Trentesaux D., Les systèmes de pilotage hétérarchiques : innovations réelles ou modèles stériles ?, *Journal Européen des Systèmes Automatisés*, Lavoisier, ISSN 1269-6935, vol. 41, n°9-10, 2007, pp. 1165-1202.
- Trentesaux D., Dindeleux R., Tahon C., A MultiCriteria Decision Support System for Dynamic task Allocation in a Distributed Production Activity Control Structure, *Int. Journal of Computer Integrated Manufacturing*, Taylor & Francis, ISSN 0951-192X, Vol. 11 n°1, 1998, pp. 3-17.
- Trentesaux D., Pesin P., Tahon C., Distributed artificial intelligence for FMS scheduling, control and design support, *J. of Intelligent Manufacturing*, vol. 11, n° 6, 2000, pp. 573-589.
- Trentesaux D., Pesin P., Tahon C., Comparison of Constraint Logic Programming and Distributed Problem Solving: a case study for interactive, efficient and practicable job-shop scheduling, *Computer and Industrial Engineering*, vol. 39, 2001, pp. 187-211.
- Wago system, innovative connections, <http://www.wago.com>, 2007.
- Wilensky U. <http://ccl.northwestern.edu/netlogo/>. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL, 1999.
- Wong T.N., Leung C.W., Mak K.L., Fung R.Y.K, Dynamic shopfloor scheduling in multi-agent manufacturing systems, *Expert Systems with Applications*, vol. 31, 2006, p. 486–494.

Zbib N, Raileanu S., Sallez Y, Berger T, Trentesaux D. From “Passive Products to Intelligent Products: The Augmentation Module Concept”. 5th CIRP-sponsored, International Conference on Digital Enterprise Technology – DET, Nantes, France, 22-24th October 2008.