We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

4

Bio-inspired approach for autonomous routing in FMS

T. Berger, Y. Sallez and C. Tahon

1. Introduction

Today's mass production strategies are unable to cope with the present needs of the manufacturing industry, which is constantly confronted with changing and increasingly complex product requirements as well as mounting pressure to decrease costs. To meet this challenge, Flexible Manufacturing Systems (FMS) must become more robust, scalable, reconfigurable, dynamic, adaptable and even more flexible. The challenge is even greater given that both production program modifications and resource failures can necessitate a partial or total reconfiguration of the FMS routing. Such reconfiguration is difficult to accomplish both because the problem is huge (combinatory explosion) and because each individual failed resource situation requires anticipating a different partial solution. In response to the FMS challenge, this chapter proposes autonomous routing of physical FMS flows.

Many international research projects focusing on the design of heterarchical (non-hierarchical) architectures have already been completed. Such architectures play a prominent role in the new control systems that dominate the field of FMS research (Duffie & Prabhu, 1996; Pujo & Kieffer, 2002). Our contribution to such decentralized control is inspired by the biological phenomenon called stigmergy, defined as insects' use of chemicals, called pheromones, to organize group activity. For example, foraging ants are known to lay down chemical trails, which are, in turn, followed by other ants who add their own odour to the trail, thus reinforcing it for future use. This stigmergic activity has an optimizing effect on performance. It allows creatures to communicate indirectly by sensing and modifying their local environment, and it is this communication that determines the creature's behaviour.

The next section of this chapter describes the concept of stigmergy. After presenting the different pheromonal characteristics upon which our bio-inspired approach is based, we describe several pheromone emulation mechanisms and some interesting insect-based methods that have been devised for various

manufacturing applications. Section 3 explains the key phases of our approach, focusing successively on the virtual pheromone-based progression of the entities and the updating of the virtual pheromones. Section 4 summarizes the results of a flexible assembly cell simulation that was conducted at the AIP-PRIMECA Center in Valenciennes. Following a description of the cell architecture and its main components and a brief presentation of the Netlogo simulation context, the qualitative and quantitative results are presented.

Section 5 outlines the advantages (adaptability, robustness) and potential disadvantages (stagnation, delay) of the stigmergic approach, and proposes solutions to the problems of stagnation and delay that can occur. Section 6 presents a real-life implementation of our approach, involving the instrumentation of moving entities and their environment. Finally, the last section of this chapter provides a brief overview of our prospective future research on self-organization.

2. An approach based on insect societies

2.1 The stigmergy concept

French entomologist Grassé (1959) introduced the term "stigmergy" to describe the mechanism by which termites coordinate their mound-building activities. In such activities, many individuals participate in a collective task, and the stimuli provided by the emerging structure are used to coordinate the individual actions. A similar mechanism is used by ants laying down pheromone trails between a food source and their nest.

Figure 1 portrays a classic experiment in entomology. Starting in the upper left-hand quadrant of the figure 1a, ants wander around their nest in search of food. Those finding food carry it back to the nest, simultaneously laying down a pheromone trail (figure 1b). Other ants, detecting the pheromones, follow the trails back toward the food. As more ants bring food to the nest, they each reinforce the chemical trail of the path they follow. Since pheromones tend to evaporate over time, the more attractive trails accumulate more pheromones and thus an advantage over the other trails (figure 1c). Over time, due to the natural reinforcement of the ants, only the shortest trail remains (figure 1d). As the experiment illustrates, this stigmergic process has a natural optimizing effect. (For more information about the history of stigmergy in the context of social insects, see (Theraulaz & Bonabeau, 1999))
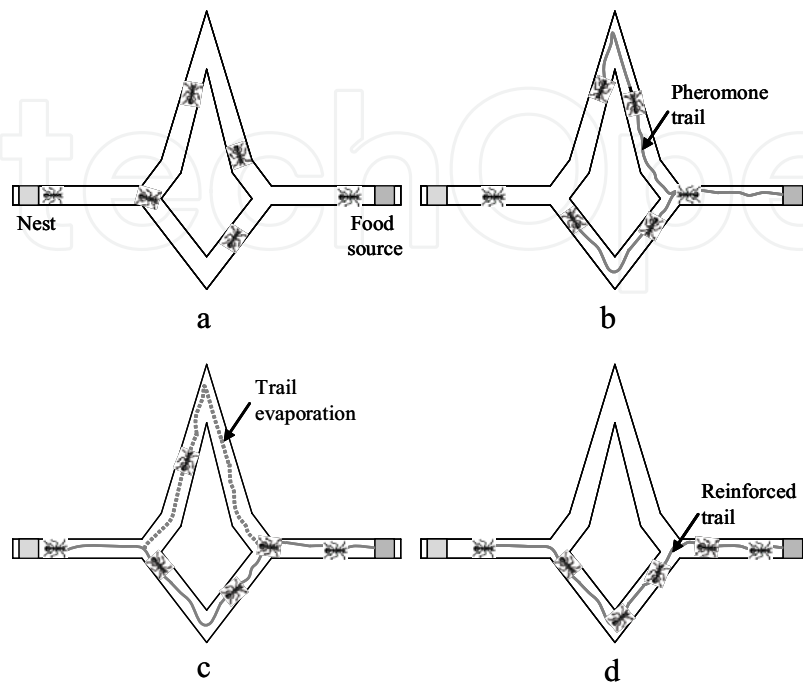
Figure 1. Stigmergy illustration

2.2 Basic operations

In the real world, three basic operations have been associated with the stigmergic process: information fusion, information removal and local information distribution. In the first, deposits from individual entities are aggregated to allow the easy fusion of information. In the second, pheromone evaporation over time is used to remove obsolete or inconsistent information. In the last, information is provided according to pheromone diffusion in the immediate (local) neighbourhood.

In all of these operations, the pheromone field has three main characteristics:

1. Independence: The sender of a pheromone message does not know the identity of the potential receiver and does not wait for any acknowledgement, which makes pheromone use very effective for communication within large populations of simple entities.
2. Local management: Because pheromone diffusion falls off rapidly with distance, pheromone interaction remains local, thus avoiding the need for centralized interaction management.
3. Dynamism: The continuous cycles of reinforcement and evaporation act respectively to integrate new information and to delete obsolete information.
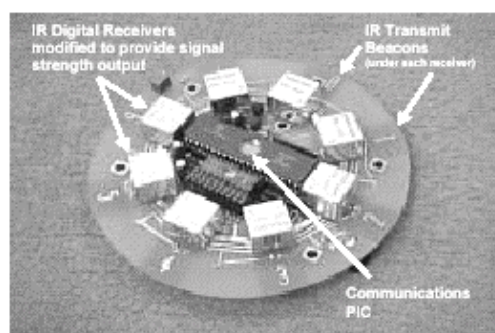
2.3 Pheromone emulation

A variety of different approaches to pheromone emulation have been developed. The three most common types are described below:

<u>Common memory</u> In many studies (e.g., Dorigo & Colombetti, 1998), artificial ants cooperate via a common memory that serves the same purpose as the pheromones deposited by real ants. In this common memory, an artificial pheromone is created and accumulated (updated) via a learning mechanism during runtime.
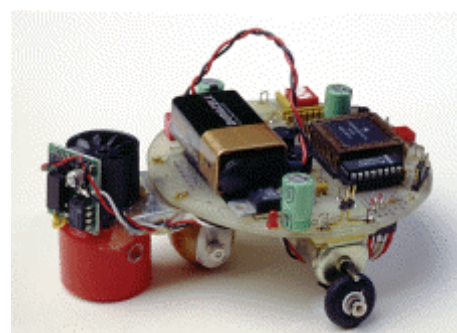
<u>Embedded implementation</u> Using signals transmitted from robot to robot, Payton et al. (2001) have implemented "virtual" pheromones that are sustained on board. Simple beacons and directional Ir sensors are mounted on the robots (figure 2a), with the virtual pheromones attached to the robots rather than laid down in the environment. This particularity is necessitated by the application: guidance of a rescue team in an unfamiliar building.

<u>Direct environmental marking</u> This kind of pheromone emulation can be performed in two ways: using real "pheromones" and using deposit tags.

- Real pheromones have been used by researchers in Australia (Russell, 1995) (figure 2b) and in Israël (Wagner et al., 1995) to emulate ant behaviour by creating robots capable of laying down and detecting chemical trails (camphor or thinner).
- Deposit tags, such as the Intelligent Data Carrier (IDC), have been developed by Japanese researchers (Kurabayashi, 1999). The IDC system is composed of reader/writer units attached to mobile robots and tags that are carried and located by the robots. These tags are analogous to pheromones in that they store the information used to guide the robots (figure 2c).

a) Transceiver for virtual pheromones   b) Robot chemical sensing
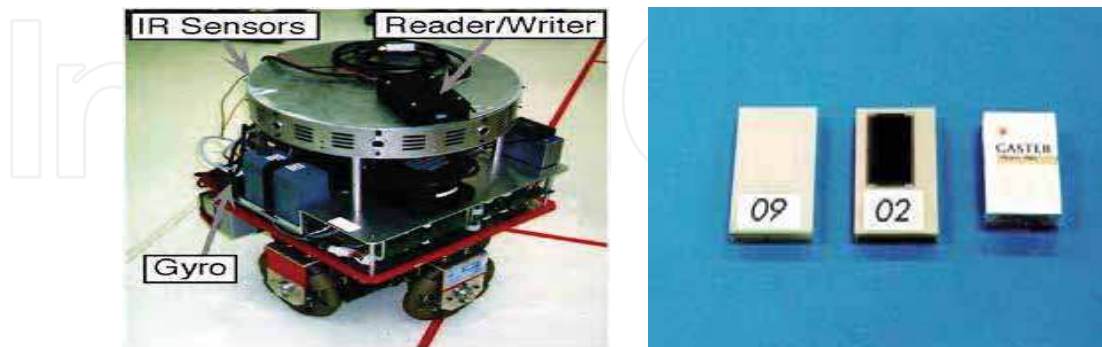
c) Intelligent Data Carrier (robot and tags)

Figure 2. Examples of existing pheromone emulation hardware

2.4 FMS applications

The first experiments related to the industrial use of stigmergy were con-
ducted in the early 1980s by Deneubourg et al. (1983), who simulated "ant-like
robots". Since then, many researchers (e.g., Ferber, 1995; Arkin, 1998; Dorigo &
Colombetti, 1998) have applied this concept when studying robot collectives
and working to solve optimization problems (e.g., Travelling Salesman Prob-
lems, Network Routing for telecommunications and the Internet). Based on the
ant foraging analogy, Dorigo et al. (1999) developed the Ant Colony Optimiza-
tion (ACO) metaheuristic, a population-based approach to solving combinato-
rial optimization problems. The basic idea behind ACO is that a large number
of simple artificial entities can be used to build good solutions to hard combi-
natorial optimization problems via low-level communications. The ACO ap-
proach can be applied to almost any scheduling problem, such as job shop
scheduling and vehicle routing, for example.

Researchers have also applied the stigmergy concept to specific situations in
manufacturing control systems:

- Parunak et al. (2001) emphasize the importance of the environment in
  agent systems, in which information flows through the environment com-
  plement classic message-based communications between the agents. In
  this study, the environment is computational, and agents moving over a
  graph are used to study manufacturing company supply networks. The

authors focus on the dynamics that emerge from the interactions in multi-agent systems; these dynamics are analyzed using methods inspired by statistical mechanics.

- Brückner (2000) applies the stigmergy concept to manufacturing control, and his application is supported by an agent-system approach. He presents an extensive set of guidelines that can be used to design synthetic ecosystems. In this study, different types of agents are used to model the various elements (e.g., resources, part flow and control units) involved in routing a car body through a Mercedes Benz paint shop. Brückner's work is based on the PROSA reference architecture (Wyns, 1999).

- Peeters et al. (1999) and Hadeli et al. (2004) both propose a pheromone-based control algorithm with a bottom-up design. Like Brückner (see above), Peeters et al. based their work on the PROSA reference architecture (Wyns, 1999). (Those interested should consult the Mascada-WP4-Report (1999) for a description of both the agents and the pheromone life-cycle in the routing of a car body through a paint shop, this one at Daimler-Chrysler). Hadeli et al. (2004) emulate a simple flexible manufacturing system characterized by dynamic order arrival, probabilistic processing time, and several disturbances (e.g., machine breakdowns), with the objective of evaluating the possibility of creating short-term forecasts based on agent intentions.

3. Description of our bio-inspired approach

Our approach to FMS is based on the behaviour of biological systems, such as ant colonies (Deneubourg et al., 1983; Di Caro & Dorigo, 1998). An FMS can be seen as a network of nodes that are interconnected by uni/bi-directional paths on which mobile product entities navigate. Each of these entities must obtain a variety of services from resources (service stations) located on the nodes. These autonomous entities move from one node to another until they reach a destination node, where the desired service can be obtained. One after another, the entities choose a destination and the appropriate path to reach it, using the data stored on each node. Typical entity behaviour is portrayed in schematic form below (figure 3).
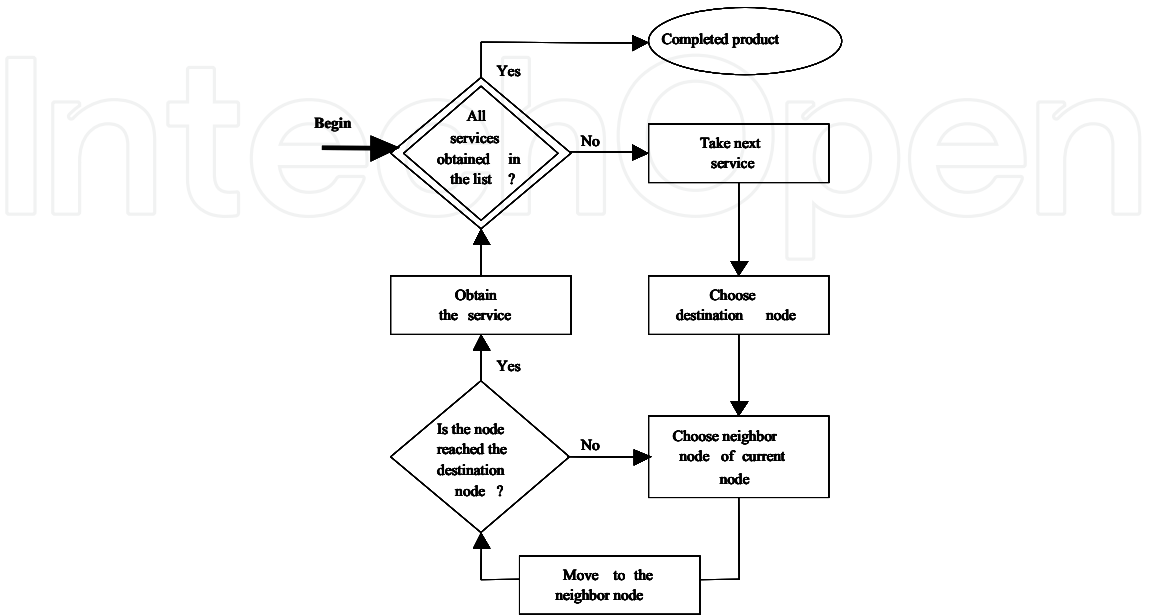
Figure 3. Typical entity behaviour

3.1 Entity progression based on virtual pheromones

In analogy with biological systems, *P* coefficients characterize the pheromone rate on different trails/paths. Based on the information stored on the current node, which includes the coefficients associated with each destination node, the entity chooses the best route from its current location to the destination node.

Let P_{dn}, on node n_k, represent the preference for the neighbour node n_n that comes with a short traversal time to move from the current node n_k to destination n_d via node n_n, which must belong to the n_k neighbour node set.

In the following notation:

$$P_{dn} \in [0;1], \ P_{dj} > P_{dp} \tag{1}$$

implies n_j allows the entity to reach n_d more rapidly than n_p, where n_j and n_p are neighbours of n_k.

$$Vn_k \Leftrightarrow \{ n_k \text{ neighbour} \}; \ n_n \in Vn_k \tag{2}$$

After standardization:

$$\sum_i P_{di} = 1; i \in Vn_k \qquad (3)$$

The best neighbour is n_c, such that:

$$P_{dc} = \max(P_{di}); i / n_i \in Vnk \qquad (4)$$

Still, since both chance and diversity are important adaptation mechanisms in natural biological systems, the choice of n_c is not totally deterministic. In fact, a sort of "roulette wheel", based on the same principles as the one found in a casino, is used to choose the next neighbour. The different P_{dn} coefficients serve as weights to build a 100-sector roulette wheel. Higher P_{di} coefficients increase the likelihood that neighbour n_i will be chosen. To avoid one neighbour becoming predominant, the minimal P_{di} coefficient value is 0.01 (at least one chance in 100). This "roulette wheel" principle is presented in figure 4.
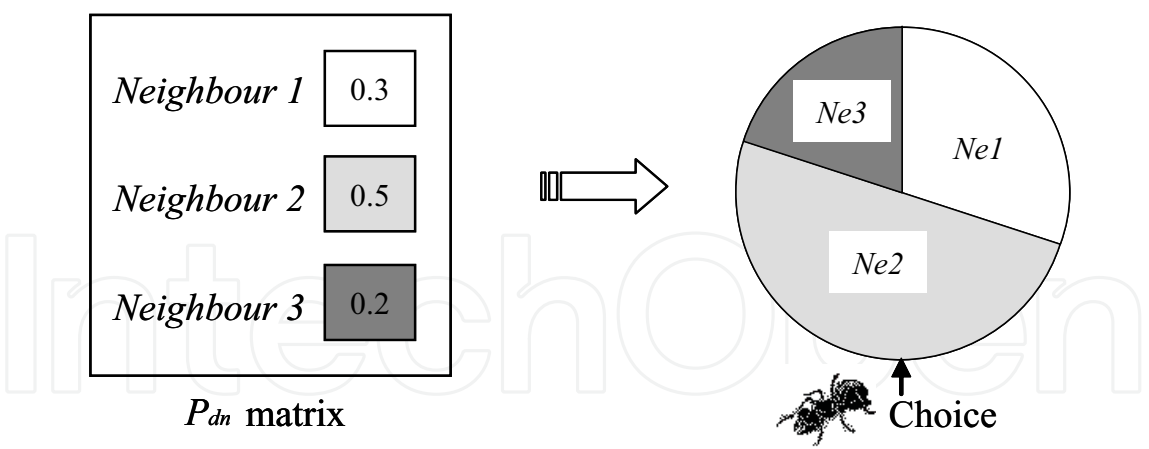

Figure 4. Roulette wheel illustration

3.2 Virtual pheromone updating

While moving, each entity stores data in its embedded memory. The memory records the entity's path through the nodes n_k, including crossing time. Like real ants, which lay down a pheromone trail when returning to the nest, every

time an entity reaches a destination node, a fictitious entity retraces its path virtual*y*, using the information contained in the embedded memory to update coefficients at each node that was crossed.

		N4	N5	N6	N7	N8
Neighbor 1	N4	PN4, N4 **(1)**	PN5, N4 **(0)**	PN6, N4	PN7, N4	PN8, N4
Neighbor 2 P_{dn}	N5	PN4, N5 **(0)**	PN5, N5 **(1)**	PN6, N5	PN7, N5	PN8, N5

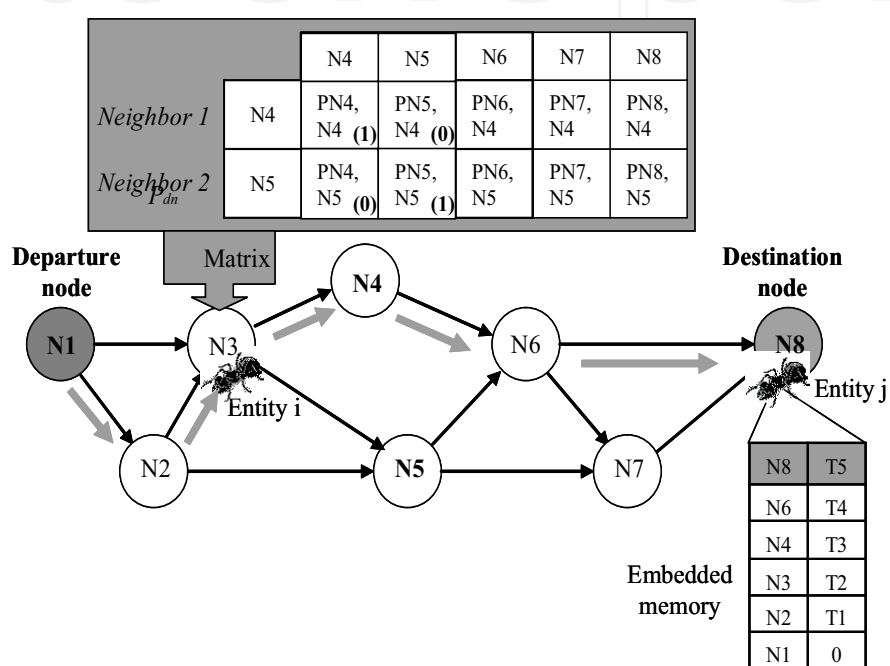| N8	T5
N6	T4
N4	T3
N3	T2
N2	T1
N1	0

Figure 5. The laying down of pheromone information

Figure 5 shows an example of the embedded memory for an entity j, which has just arrived at a destination node. Each node n_k includes:

- a P_{dn} matrix, whose columns provide all possible destinations n_d and whose rows show all existing neighbours Vn_k; and
- a matrix that contains μ_{dk}, the mean time needed to go from n_k to n_d for all possible destinations (and if used, the standard deviation σ_{dk}).

T_{dk} is the time span needed to go from n_k to n_d. When the fictitious entity arrives at node n_k, T_{dk} and the mean μ_{dk} of the previous T_{dk} are compared. The P_{dn} matrix is then updated by incrementing the coefficient P_{dc} (the possibility of choosing neighbour n_c when the destination is n_d) and decrementing other coefficients P_{do}.

A reinforcement value r (in our case r = 0.2) is used as follows:

$$P_{dc} \Leftarrow P_{dc} + \text{r} * (1 - P_{dc}) \qquad\qquad (5)$$

Coefficients P_{do} for the destination n_d of the other neighbours n_o are negatively reinforced through a process of normalization.

$$P_{do} \Leftarrow P_{do} - \text{r} * P_{do}, \; n_o \in V_{n_k} \text{ with } n_o \neq n_c \qquad\qquad (6)$$

For a more detailed discussion about how r is chosen, please refer to the literature on "reinforcement learning", specifically the books by Dorigo & Stützle (2004) and Sutton & Barto (1998).

A more sophisticated approach to updating the P coefficients could integrate the standard deviation σ_{dk} into the updating process. In such an approach, the comparison of T_{dk} and μ_{dk} would be valid only if μ_{dk} were sufficiently stable in terms of the σ_{dk} of the previous T_{dk}. Three situations are possible:

- given a stable μ_{dk} value and $T_{dk} < \mu_{dk}$, increasing P_{dc} would reinforce n_c,
- given a stable μ_{dk} value and $T_{dk} \geq \mu_{dk}$, decreasing P_{dc} would result in n_c being ignored,
- given an unstable μ_{dk} value, adjustments would need to be made to stabilize it.

4 An FMS routing simulation

4.1 The AIP FMS cell

A flexible assembly cell was simulated at the Valenciennes AIP-PRIMECA Center. This cell was composed of seven workstations W_i placed around a flexible conveyor system, which insured a flexible flow of pallets to each workstation. The conveyor system is based on Montech's Montrac system (Montech, 2005). Montrac is a monorail transport system that uses self-propelled shuttles to transport materials on tracks (figure 6). Each shuttle is individually controlled and equipped with a collision-avoidance optical sensor.
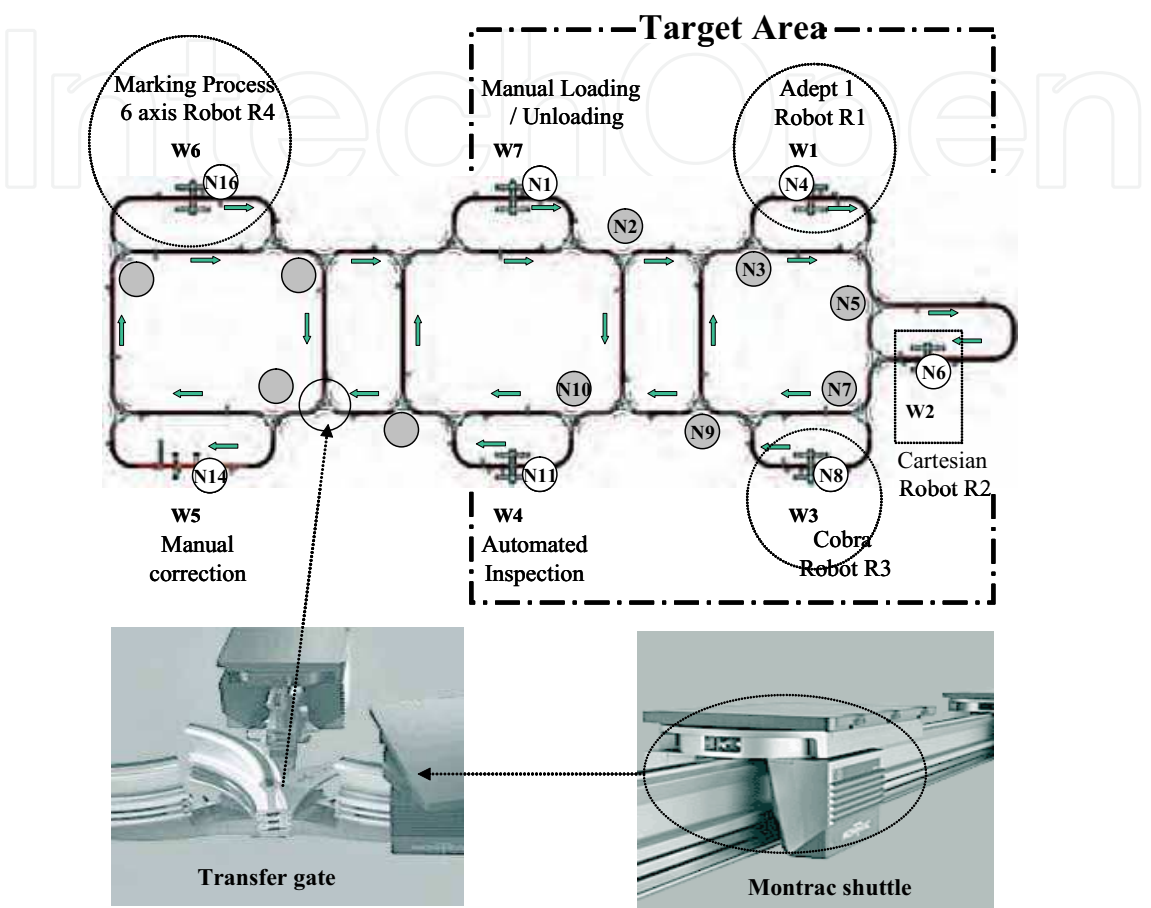
Figure 6. Schematic view of the flexible cell with inset detailed images of the transfer gate and shuttle

Eleven of the major nodes in the cell shown in figure 6 were considered in this simulation:

- The nodes/stations in white (N1, N4, N6, N8 and N11) are possible destination nodes where services can be obtained.
- The nodes/stations in gray (N2, N3, N5, N7, N9 and N10) are divergent transfer gates, from which shuttles can obtain the information available about the destinations in order to make their routing decisions.

The other transfer gates (neither white nor grey) appearing in figure 6 were not taken into account in the simulation. They were only used to connect convergent tracks when no routing decisions were required.

4.2 The context of the simulation

Given the specifications of the cell described in the previous section, we chose the NetLogo platform (NetLogo, 2006) to simulate our approach. The NetLogo platform offers an agent-based parallel modelling and simulation environment. Mainly used to simulate natural and social phenomena, it is particularly well suited to complex system modelling. With Netlogo, each entity can be described as an independent agent interacting with its environment. All agents operate in parallel on a grid of patches (cellular world), and each agent can read and modify some of the attributes linked to the patches in its proximity. The behavioural rules defined for the agents make it possible to describe agent-environment interaction, which is very important when simulating the stigmergic process.

4.3 Results

4.3.1 Qualitative results

The preliminary results of the first simulation clearly illustrate the overall adaptability of the system when faced with small disturbances. Figure 7 shows the right part of the cell as depicted with a NetLogo simulator. In the figure 7a, N1 and N11 are the departure and destination nodes, respectively; and the location of the entities is indicated with small arrows.
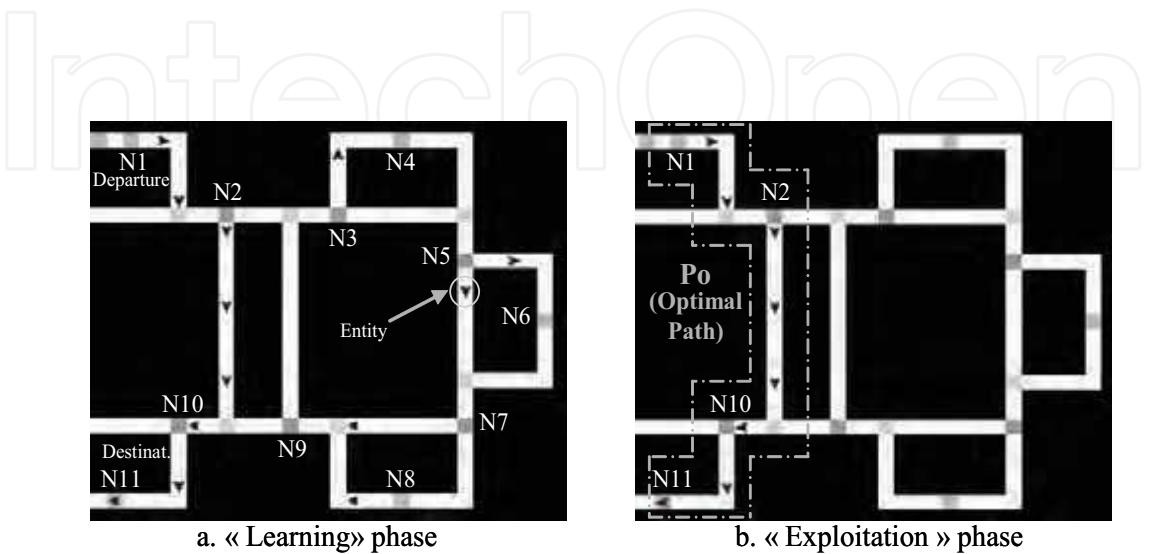

a. « Learning» phase   b. « Exploitation » phase

Figure 7. Netlogo simulation results

The simulation can be divided in two phases: the "learning" phase and the "exploitation" phase.

-   The learning phase is shown in figure 7a. First, the entities traveled randomly along the different paths of the network, and all the P coefficients were equal on the different nodes.
-   The exploitation phase began when the P coefficients were stable, which happened when all the paths to the different destination nodes had been sufficiently explored (figure 7b).

The pivotal point, at which the learning phase became the exploitation phase, occurred when, in accordance with the principle of stigmergy, the optimal path Po (N1-N2-N10-N11) emerged through reinforcement (figure 7b). However, the stigmergic process is not static. Faced with disturbances (e.g., flow reduction) that affected the fluidity of the path N2-N10, the entities travelling on this path began to perform poorly, and the appeal of this path decreased. The decreased appeal of path Po increased the appeal of path NPo (N1-N2-N3-N5-N7-N9-N10-N11), even though this path was originally non-optimal (figure 8). This dynamic response is a classic display of the natural routing reconfiguration capacity of the stigmergic approach.
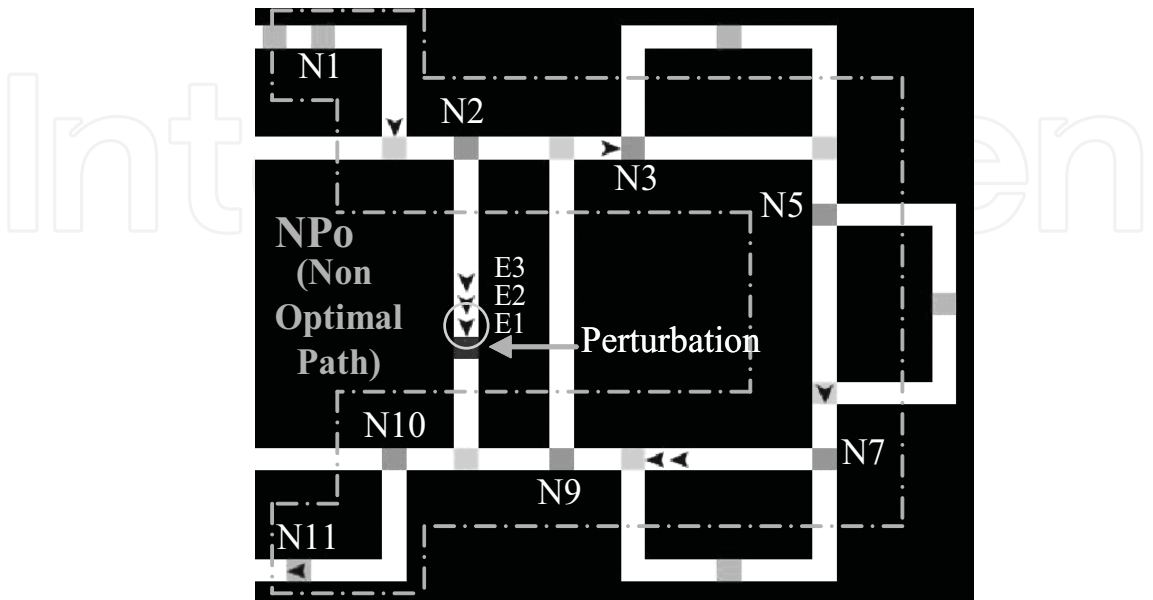

Figure 8. Reconfiguration capacity of the stigmergic approach

4.3.2 Quantitative results for the target area

In this second simulation, the departure and destination nodes were changed to N1 and N8, respectively. Figures 9, 10 and 11 show the evolution of the coefficients, which illustrates the advantage of passing though the current node's neighbours to reach the destination node N8.
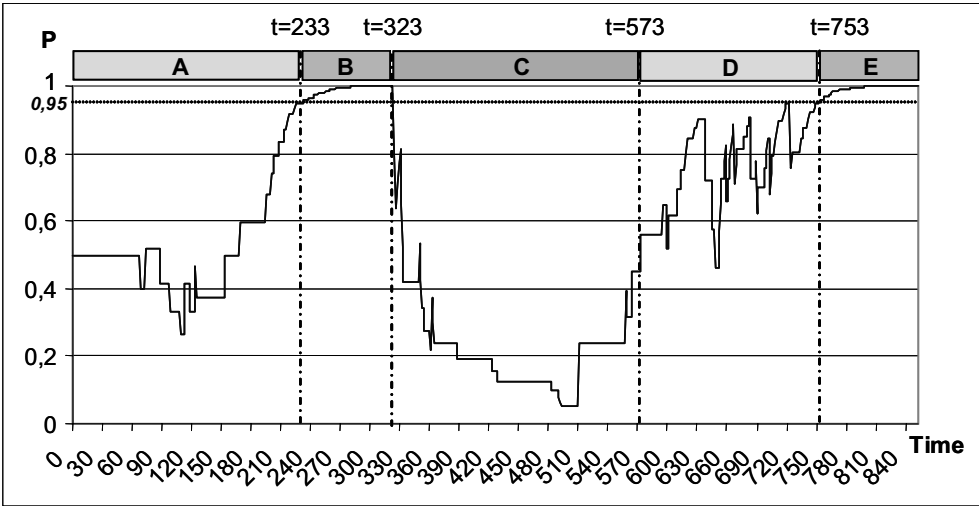

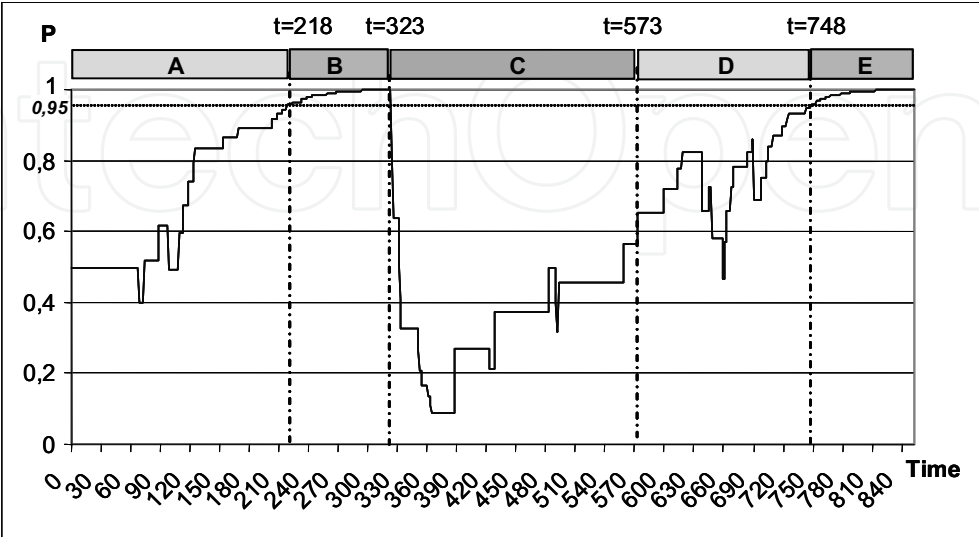Figure 9. Evolution of coefficient P on N2, showing the advantage of going through N3 to reach N8.

Figure 10. Evolution of coefficient P on N3, showing the advantage of going through N5 to reach N8.
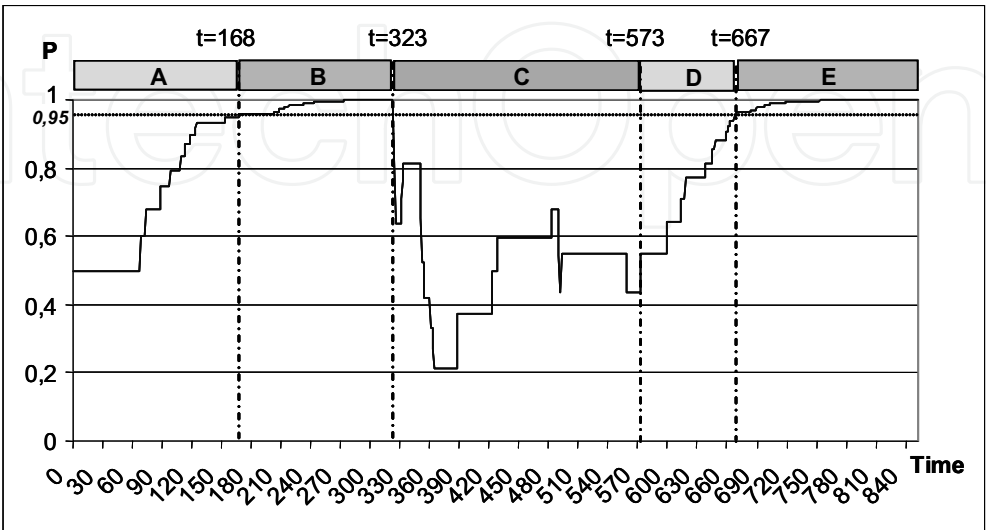
Figure 11. Evolution of coefficient *P* on N5, showing the advantage of going through N7 to reach N8.

The curves depicted in all three figures can be divided into 5 zones:

- Zone A depicts the initial operation mode, in which the system goes through a learning phase.
- Zone B depicts the "exploitation" phase, during which the system can be considered stable.
- Zone C depicts the system during a period of perturbation.
- Zone D depicts the period after the perturbation, in which the system goes through a second learning period in order to readapt/reconfigure itself.
- Zone E depicts the phase that comes about when the system is once again stable.

The system in zone A: During the initialization stage, the coefficients of both destinations are set at 0.5 until they can be updated after the entities have reached the destination node (N8). Starting at node N1, two paths are possible. In figure 9, for instance, looking at time t=60, a change in the coefficient *P* can already be observed at N2. This underlines the advantage of passing through N3 to reach N8, because it means that the first entity reached its destination at t=60, and the entity's embedded memory provided the feedback needed to update the coefficients.

The coefficient P characterizes the "optimal" path, deduced from the learning phase in which one of the possible paths passed through N2, N3, N5, N7 and N8. Consequently, there is an increase in coefficients over time. The permanent operating regime is reached at a coefficient value of 0.95. As shown in figure 11, for instance, the closer the entity is to the destination, the more quickly the coefficient reaches the value 0.95. Conversely, the farther away the entity is from the destination, the more slowly the value 0.95 is reached (figures 9 and 10). Figures 9 and 10 also show the values of the coefficients presented tend to oscillate. Depending on the topology of the network, one possible explanation for this oscillation is that the further the entity is from the destination, the larger the number of possible paths. The wider range of choices leads to more attempts to reach N8 along less efficient paths, which hinders the updating process and thus decreases the coefficient's value. This inefficiency is corrected by those entities that reach N8 more quickly, thus reinforcing the shorter paths by increasing the value of the coefficient.

The system in zone B:
Given that all the coefficients for the path from N2 to N8 have reached the value of 0.95, the learning process is presumably over. Therefore, all entities will tend to choose the same path until disturbances occur.

The system in zone C:
The smooth progression on the path from N2 to N8 (passing through N3, N5 and N7) is disrupted by an unexpected flow reduction in the path from N5 to N7, just after N5. In reaction, the system adapts itself, "forgetting" the previously optimal path by decreasing the path's coefficients.

The system in zone D:
The flow reduction has now disappeared, and following a second learning period, the path to N8 through N2, N3, N5 and N7 once again becomes the optimal choice. The closer the entity comes to the destination node, the more quickly the path coefficients stabilize themselves around the value of 0.95. In this way, the coefficients that demonstrate this path's advantage are strengthened.

The system in zone E: Entities can once again use the newly optimal path.

4.3.3 Quantitative results overall

Table 1 shows the results for all destination nodes (N4, N6, N8, N11, N14 and N16) in the cell, given a departure node N1 (see figure 6). These results are based on the average for 100 simulations. The information is presented in 5 columns: Dest Node, N entities, N loops, N straight, and Time.

- "Dest. Node" shows the identifier of the destination node.
- "N entities" gives the average number of entities arriving at destination. These entities update the P coefficients and thus help to determine the optimal path.
- "N loops" provides the average number of entities that have travelled across a loop before reaching the destination node.
- "N straight" shows the average number of entities that reached the destination node via the optimal path.
- -"Time" indicates the delay (in units of simulated time) needed for the optimal path to emerge. This delay represents the moment when all the coefficients on the optimal path are greater than 0.95.

Dest. Node	N. entities	N. loops	N. straight	Time
N4	12.00	0.87	11.13	111.61
N6	23.55	4.16	14.93	194.79
N8	38.05	8.70	18.58	278.87
N11	12.01	0.01	10.07	120.27
N14	20.16	3.93	8.71	208.55
N16	38.11	12.12	11.60	325.54

Table 1. Summary of the overall results

Logically, the number of entities and the time needed for the optimal path to emerge increases with the distance between the departure and destination nodes. The main differences in the simulation results are due to the presence of loops in the network (see figure 6). These loops delay the convergence of certain coefficients. Using a 800 MHz PC, the CPU time needed to determine the optimal path ranges from 1 to 6 seconds, depending the destination.

5 Advantages and disadvantages of the stigmergic approach

The results described above demonstrate the advantages of our bio-inspired approach. First, the entities 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 disturbances, by seeking out new paths that bypass the disturbance but still lead to the desired destination. These capacities indicate that the approach offers a good level of adaptability and robustness in the face of environmental fluctuations. These qualities are common in classic biological systems based on stigmergic principles (e.g., ant colonies, termite mounds).

However, there are two potential problems, which can be disadvantageous: *stagnation*, once the routing network reaches its convergence point, and the *delays* caused by the traffic jams.

Stagnation occurs when the routing network reaches its convergence point. The optimal path Po is now chosen by all ants, which recursively reinforces the preference for Po. Unfortunately, this reinforcement has two negative consequences, in addition to the positive ones described above: 1) the optimal path becomes congested, while 2) the likelihood that another path will be chosen to avoid the congestion is significantly decreased.

During the simulation, frequent disturbances on the path N2-N10 provoked a decrease in the corresponding coefficient P at node N2, resulting in a preference for the path NPo (see figure 8). Little by little, the optimal path Po is "forgotten". However, this problem can be resolved. In their survey of ACO systems, Sim & Sun (2003) describe several mechanisms that can be used:

- Evaporation: This mechanism has been widely used in many studies (e.g., Parunak et al., 2001; Brückner, 2000). Evaporation prevents pheromones from concentrating on certain paths and allows new routing solutions to be explored. This approach is inspired by real ant colonies.
- Aging: With the aging mechanism, older entities deposit less and less of the pheromone as they travel from node to node. This solution, often used in conjunction with evaporation, is based on the fact that "old" entities are less successful at finding optimal paths.
- Limiting: This mechanism limits the quantity of the pheromone that can be amassed on any one path. This upper limit prevents a specific path from becoming "too dominant".

The third mechanism was chosen for our approach, allowing "chance" to play a bigger role in path exploration.

Delays were quite typical in the FMS application simulated in this experiment. When the optimal path Po emerged, it became dominant, and all entities followed this path to N11. If an entity E1 got stuck on the path N2-N10, the entities that followed were also blocked (figure 12). Unfortunately, this kind of disturbance can not be detected immediately because the P coefficients are only updated after entity E1 arrives at its destination, node N11. In nature, real ants bypass the obstacle and continue on the path. However, in our simulation, the entities (shuttles) remained blocked on the tracks and could not bypass the obstacle.

To solve this problem, we introduced a repulsive field, commonly used in mobile robotics (Balch & Parker, 2002). In the real implementation, this repulsive field was created by messages exchanged between nodes. (See section 6.3, below, for more details.) With the use of a repulsive field, the P coefficient corresponding to the blocked path is temporarily ignored, and the other shuttles take a detour that brings them to the destination node. After the problem causing the obstacle has been resolved, the "old" P coefficient is restored for this path.
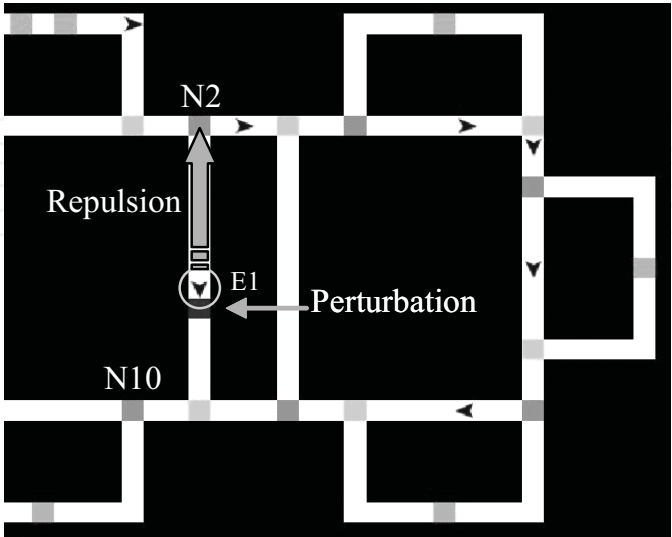

Figure 12. Repulsive effect

6. Details for a real implementation

6.1 The implementation

Following the simulation phase, a real implementation was designed and is currently being tested.

As explained in section 3, a node is assigned to each location where a decision must be made. At the current node n_k, a shuttle queries the node to gain read access to the P coefficients, which are stored in node n_k. To apply our approach, two types of equipment must be installed (see figure 13):

  - *Node instrumentation*, including a "gate controller" which works to oversee the transfer gate and to help avoid collisions, a P_{dn} matrix (see figure 5), another matrix containing the mean μ_{dk}, two data communication systems (both Ethernet and Ir), and a data processing system; and
  - *Shuttle instrumentation*, including an Ir communication system and a data processing system.
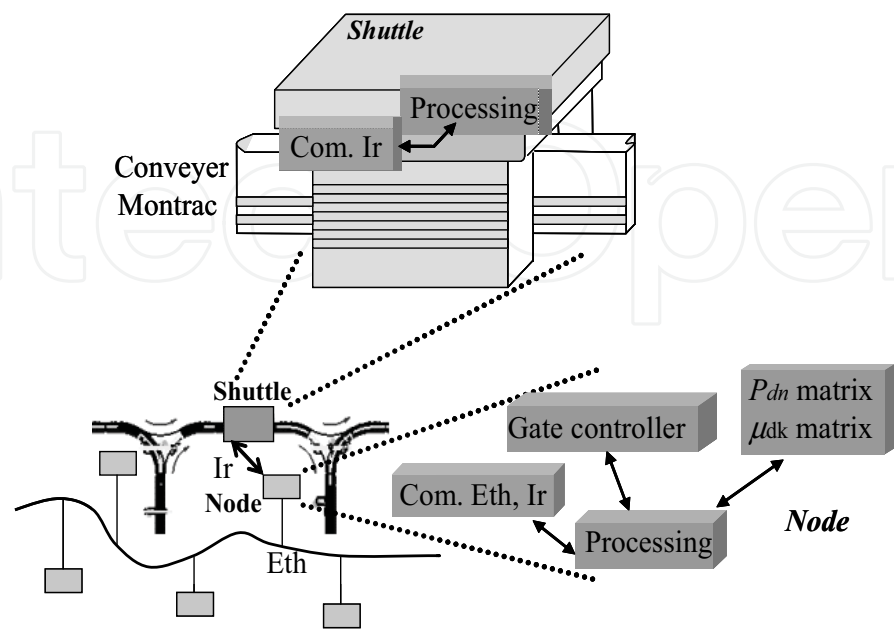

Figure 13. Equipment necessary to implement our approach

6.2 Shuttle progression

In order to identify its position, the shuttle obtains the ID node at each current node n_k through Ir communication. As required in our approach, the shuttle also obtains the P coefficients needed to allow it to choose the best neighbour node. These shuttle queries are processed by a microcontroller (Beck-IPC) located on the node. Then, via a microcontroller (Microchip Pic18F) embedded inside the shuttle, the shuttle processes the P coefficients in order to choose the best neighbour. Once the processing phase is finished, the shuttle asks to be routed towards the chosen neighbour node. The concrete local routing is performed by the "gate controller" on the node.

6.3 P updating and the implementation of the repulsive field

When a shuttle arrives at the destination node, it uploads its embedded memory. The node controller sends the data to the nodes that the shuttle passed through, and each node updates its P_{dn} matrix and μ_{dk} matrix. This inter-node communication is done through an Ethernet link.

To implement a repulsive field, the node from which a shuttle has just departed launches a timer. When the information from the shuttle arriving at the chosen neighbour is sent back to the previous node via the Ethernet link, this timer is reset. When the time delay exceeds the timeout limit, the previous node considers that the shuttle is blocked on the path to the chosen neighbour, and the P coefficient corresponding to this blocked path is temporarily ignored.

7. Perspectives for future research

To solve the delay problem in the first approach, we introduced a solution based on repulsion. A more general and interesting solution might be to couple the stigmergic approach with more reactive methods, such as those based on potential-fields. Potential field-based methods are inspired by electrostatics. They are widely used in swarm robotics to navigate mobile robots in restricted environments (Arkin, 1998; Balch & Parker, 2002). They combine attractive forces (goals) and repulsive forces (obstacles) to guide reactive robots. In fact, all entities (goals, obstacles, others robots in proximity) are considered to be attractive or repulsive forces, with each mobile robot moving under the vector sum of all the previous forces. In classic field-based approaches, the repulsive or attractive forces act only inside an area defined by fixed distances.

In their work on bionic manufacturing systems, Vaario and Ueda (1998) have also used local attraction fields to direct transporters carrying jobs to particular resources, resulting in the emergence of a dynamic schedule of the interactions between the different entities. In the context of FMS routing, the idea of bionic systems would be an interesting avenue to explore.

As seen in section 4, our simulation is bound by the topology of the AIP FMS cell. Shuttles must obligatorily follow the tracks and can be jammed by any failure of the conveyor system. To examine the possible intersection of stigmergy and potential field-based methods, we looked at an experiment similar to the line-less production system, introduced by Ueda et al. (2001).

In a line-less production system, all production entities (products, resources) can move freely on the production floor. Self-organization concepts are used to deal with fluctuations (diversity of the production demand, machine failures). The authors' application context is a car chassis welding process. Car chassis are mounted on automated guided vehicles (agvs) capable of moving around the shop floor. Mobile welding robots move towards the agvs according to the perceived attraction field. Figure 14 shows an agv with 6 sub-parts and 2 types of welding robots (A and B). Depending on the product requirements, different attraction fields can be generated. The results obtained by Ueda et al. suggest that this approach is appropriate for high-variety production and can generate high productivity (Ueda et al., 2001).
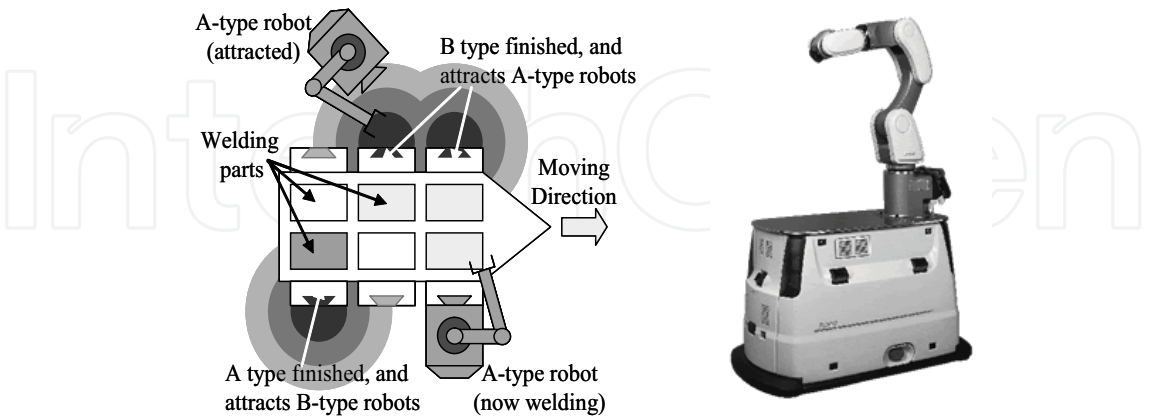
Figure 14. Agv attracting robots (Ueda et al., 2001) and mobile robot

These two complementary approaches—stigmergy and potential field-based mechanism—are dynamic local methods that don't require centralized management; both characteristics are essential when studying emergent phenomena in self-organized processes.

8. Conclusion

The stigmergic approach is an original answer to routing problems in FMS. Applied in many research fields (e.g., robotics, network routing), stigmergy offers robustness and adaptability in the face of environmental variations. The results of our simulation in the NetLogo environment highlight the robustness and adaptability of the approach, and, in fact, these qualities stem directly from our use of virtual pheromones. Although the approach is not perfect, solutions exist to remedy the problems of stagnation and delay. For example, when the stigmergic approach fails to take disturbances into account in real-time (e.g., a blocked shuttle), this lack of reactivity can be resolved by adopting complementary mechanisms, such as a repulsive field.

Our main objective is to develop adequate ways (e.g., stigmergic, field-based or hormone-based mechanisms) to support the interactions between entities in self-organized processes, such as bionic manufacturing systems. The implementation of the bio-inspired method described here is a first step towards the development of more intelligent flows in FMS.

9. References

Arkin, R.C. (1998). *Behaviour-based robotics*, The MIT Press

Balch, T. & Parker, L.E. (2002). *Robot teams: From Diversity to Polymorphism*, Natick, Massachusetts: A K Peters Ltd.

Brückner, S. (2000). Return from the Ant synthetic ecosystems for manufacturing control, Thesis Humboldt-University of Berlin, june 2000

Deneubourg, J.L.; Pasteels, J.M. & Verhaeghe, J.C. (1983). Probabilistic Behaviour in Ants : a Strategy of Errors ? *Journal of Theoretical Biology*, Vol. 105, pp. 259-271, 1983

Di Caro, G. & Dorigo, M. (1998). AntNet : Distributed Stigmergic Control for Communications Networks. *Journal of Intelligence Research*, 9, pp 317-365, 1998

Dorigo, M. & Colombetti, M. (1998). *Robot shaping: An experiment in Behaviour Engineering*, The MIT Press

Dorigo, M.; Di Caro, G. & Gambardella, L.M. (1999). Ant algorithms for discrete optimization. *Artificial Life*, Vol. 5, No 2, pp. 137-172, 1999

Dorigo, M. & Stützle, T. (2004). *Ant Colony optimization*, The MIT Press

Duffie, N.A. & Prabhu, V. (1996). Heterarchical control of highly distributed manufacturing systems, *International Journal of Computer Integrated Manufacturing*, Vol. 9, No. 4, pp. 270-281, 1996

Ferber, J. (1995). *Les systèmes multi-agents*, InterEditions

Grassé, P.P. (1959). La reconstruction du nid et les coordination inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. La théorie de la stigmergie : essai d'interprétation du comportement des termites constructeurs, Insectes Sociaux, Vol. 6, pp. 41-83, 1959

Hadeli, T.; Valckenaers, P.,Kollingbaum, M. & Van Brussel, H. (2004).Multi-agent coordination and control using stigmergy. *Computers in industry*, Elsevier Science, Vol. 53, pp. 75-96, 2004

Kurabayashi, D. (1999). Development of an Intelligent Data Carrier (IDC) system and its applications, *Proceedings of 4th Int. Symp. on Artificial Life and Robotics*, pp. 34-39, 1999

Mascada-WP4 (1999). WP4 Report : ACA (Autonomous Co-operating Agents) Framework for Manufacturing Control Systems, 1999

Montech (2005). website: http://www.montech.ch/montrac/content/

NetLogo (2006). website: http://ccl.northwestern.edu/netlogo/

Parunak, H.V.D.; Brueckner, S. & Sauter, J. (2001). 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

Payton, D.; Daily, M., Estkowski, R., Howard, M. & Lee, C. (2001). Pheromone Robotics. *Autonomous Robots*, Vol. 11, No. 3, pp. 319-324, Kluwer Academic Publishers, Norwell (MA)

Peeters, P.; Van Brussel, H., Valckenaers, P., Wyns, J., Bongaerts, L., Heikkilä, T. & Kollingbaum, M. (1999). Pheromone based emergent shop floor control system for flexible flow shops, *Proceedings of International Workshop on Emergent Synthesis (IWES'99)*, Kobe, Japan, Dec. 6-7, 1999

Pujo, P. & Kieffer, J.P. (2002). *Méthodes du pilotage des systèmes de production*, Traité IC2, série Productique, Hermes Lavoisier

Russell, R.A. (1995). Laying and sensing odor markings as a strategy for assisting mobile robot navigation tasks. IEEE Robotics and Automation Magazine, pp. 3-9

Sim, K.W. & Sun, W.H. (2003). Ant Colony Optimization for Routing and Load-Balancing: Survey and New Directions. *IEEE Trans. On Systems, Man and Cybernetics,* Vol. 3, No. 5, pp. 560-572, sept. 2003.

Sutton, R.S. & Barto, A.G. (1998). *Reinforcement Learning,* The MIT Press.

Theraulaz, G. & Bonabeau, E. (1999). A brief history of stigmergy. *Journal of Artificial Life*, Vol. 5, No 2, pp. 97-116, 1999

Ueda, K.; Hatono, I., Fujii, N. & Vaario, J. (2001). Line-Less Production System Using Self-Organization: A Case Study for BMS. *Annuals of CIRP*, Vol. 50, No. 1, pp. 319-322, 2001

Vaario, J. & Ueda, K. (1998). An emergent modeling method for dynamic scheduling. *Journal of Intelligence Manufacturing*, Vol. 9, pp. 129-140.

Wagner, I.A.; Lindenbaum, M. & Bruckstein, A.M. (1995). Distributed Covering by Ant-Robots Using Evaporating Traces, *IEEE Transactions on Robotics and Automation*, Vol.15, No. 5, pp. 918-933, 1995

Wyns, J. (1999). Reference architecture for Holonic Manufacturing Systems – the key to support evolution and reconfiguration, Ph.D. Thesis K.U. Leuven, 1999

Manufacturing the Future

Edited by Vedran Kordic, Aleksandar Lazinica and Munir Merdan

ISBN 3-86611-198-3

Hard cover, 908 pages

Publisher Pro Literatur Verlag, Germany / ARS, Austria

Published online 01, July, 2006

Published in print edition July, 2006

The primary goal of this book is to cover the state-of-the-art development and future directions in modern manufacturing systems. This interdisciplinary and comprehensive volume, consisting of 30 chapters, covers a survey of trends in distributed manufacturing, modern manufacturing equipment, product design process, rapid prototyping, quality assurance, from technological and organisational point of view and aspects of supply chain management.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

T. Berger, Y. Sallez and C. Tahon (2006). Bio-Inspired Approach for Autonomous Routing in FMS, Manufacturing the Future, Vedran Kordic, Aleksandar Lazinica and Munir Merdan (Ed.), ISBN: 3-86611-198-3, InTech, Available from: http://www.intechopen.com/books/manufacturing_the_future/bio-inspired_approach_for_autonomous_routing_in_fms