Evaluation of Emergent Structures in a "Cognitive" Multi-Agent System based on On-line Building and Learning of a Cognitive Map

Abdelhak Chatty[1,2], Philippe Gaussier[2], Ilhem Kallel[1], Philipe Laroque[2], Florence Pirard[2]
and Adel M. Alimi[1]

[1]*REGIM: REsearch Groups on Intelligent Machine,*
University of Sfax, National School of Engineers (ENIS), Sfax, Tunisia
[2]*ETIS: Neuro-cybernetic Team, Image and Signal Processing,*
Cergy-Pontoise University, National School of Electronics and its Applications (ENSEA), Paris, France

Keywords: Cognitive Multi-Agent System, Cognitive Map, Emergent Structures, Optimization.

Abstract: This paper tries to analyze and evaluate emergent structures in a multi-agent system which is able to resolve the warehouse location problem. These emergent structures allow agents to optimize their planning time and to improve their adaptive behavior in an unknown environment. In our multi-agent system, each agent is based on an on-line building and learning of its own cognitive map. It alters the positive impact of individual behavior in the improvement of the overall performance of the system. We also suggest the evaluation of the emergent structures by comparing the performance of our multi-agent system with a linear programming approach. A series of simulations enables us to discuss and validate our system.

1 INTRODUCTION

To produce global emergent behaviors for multi-agents system (MAS) (Simonin et al., 1998; Billing, 2010; Pepin et al., 2009; Chatty et al., 2011; Mathieu and Secq, 2012) the interaction between agents based on stigmergy needs not to be complex (Mataric, 1992; Brooks, 1994; Bonabeau and Theraulaz, 1994). Based on this idea, researchers have been able to design a number of successful algorithms in the field of self-organized deposit of objects: (Deneubourg et al., 1990) has proposed a model relying on biologically plausible assumptions to account for the phenomenon of the clustering of dead bodies by ants. (Gaussier and Zrehen, 1994) showed that acting on objects simplifies the reasoning needed by MAS and allows the deposit of scattered objects. The question which poses itself to us is thus how to evaluate the emergent structures of MAS in the field of deposit systems? We believe that stimergy-based MAS, where agents are not aware of all of the informations regarded with the environment, prove to be adaptive. However the MAS do not guarantee the best solution, not even always a good one. As an alternative, mathematical programming approaches, such as linear programming (LP) are widely used as an optimization tool (Matsui and Matsuo, 2012) since they are able to compute the best possible configuration of a constrained system. However the distributed computing requires the overall knowledge of the system, including the information that is difficult to obtain (such as the cost of retrieving informations, the initial number of agents so on). The aim of this article is to analyze and to evaluate the emergent structures and their influence in a MAS to resolve a warehouse location problem. To prove the effectiveness of the emergent structures, we suggest comparing our MAS approach and the results provided by the LP approach. The paper is organized as follows: section 2 describes the neural network (N.N). Section 3 presents the behaviors of the cognitive agents. Section 4 and 5 are devoted respectively to the description and to the analysis of the relevance of the emerging warehouses. Section 6 evaluates and compares the performance of the multi-agent system approach with the linear programming approach on an optimization problem. The final section concludes the paper.

2 THE N.N ARCHITECTURE

Several MAS approaches for navigation have been proposed (Prodan et al., 2012; Portugal and Rocha,

2012). Based on a bio-inspired architecture, we present in this part a cognitive multi-agent system (CMAS) in which each agent is able to navigate, learn and create its own cognitive map on-line, whose structure depends on their own experience of the environment in which they live. Starting from neurobiological hypotheses on the role of hippocampus in the spatial navigation, several works (O'Keefe and Nadel, 1978) revealed special cells in the rats hippocampus that strike off when the animal is at a precise location. These neurons have been called place cells (PC). (Martinet et al., 2011) also proposed a model of a cognitive map in the prefrontal cortex, which is based on cortical columns. In our model, we do not directly use PC to navigate, plan or construct a map, we rather use neurons called transition cells (TC) (Gaussier et al., 2002). This kind of cell type represents the basis of the neurobiological model of temporal learning sequences in the hippocampus (Lubenov and Siapas, 2009). A transition cell encodes for a spatio-temporal transition between two PCs consecutively winning the competition, respectively at time t and δt. The set of the PCs and the TCs constitute a non-cartesian cognitive map. The interest of using transition cells is that their association with an action is univoque and quite straightforward. There is no need for an external algorithm to extract the action from the cognitive map. To develop our CMAS we will take inspiration from the model of (Gaussier et al., 2002). A schematic view of the architecture of our CMAS is shown in figure 1.
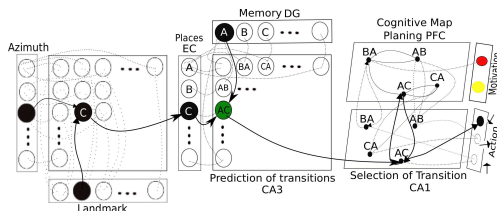

Figure 1: Model of hippocampo-cortical for the building of an agent cognitive map.

To create the PC, the agent takes, a visual panorama of the surrounding environment. The views are processed to extract visual landmarks. After learning these landmarks, a visual code is created by combining the landmarks of a panorama with their azimuth. This configuration serves as a code for PCs. During exploration of the environment, the cognitive map is gradually created when the agent moves from one place to another. The equations that govern learning in the cognitive map is $dW_{ij}^{CC}(t)/dt = T(t).((\gamma - W_{ij}^{CC}).X_i^C(t).X_j^C(t) - W_{ij}^{CC}(t).(\lambda 1.X_j^C(t) - \lambda$. $T(t)$ is a binary signal (0 or 1) which is active when a transition is made (moving from one place to another). This signal controls the learning of recurrent connec-

tions W^{CC}. Each agent has motivations related to the kind of plants. When the corresponding motivation is riggered, the agent has to reach a plant allowing the satisfaction of its needs. γ is a parameter less than 1 which regulates the distribution of the motivation activity on the map. λ1 and λ2 are parameters of respectively active and passive oblivion respectively on the recurrent connections. $S(t)$ is a signal marking the satisfaction of a goal (for example the plant discovery). This signal controls the learning of synaptic connections between neurons in W^{MC} motivations activity X^M and neurons of the cognitive map of activity X^C where the equation is $dW_{ij}^{MC}(t)/dt = S(t), for_{i,j} = arg\_max_{k,l}(X_l^C(t).X_K^M(t))$. After having explored the environment, the agents are able to predict, in each position the locations directly reachable. This model has been used in Cognitive Multi-Robot System navigation tasks (Chatty et al., 2012) in which where the agents were able to build their cognitive maps and learn how to move towards various goals in an unknown environment.

3 THE COGNITIVE-MAS

We suppose an economic environment composed by several Animats and three plants with an abstract representation namely A, B and C. The CMAS are motivated by the simulation of three types of needs related to the three plants and each need can be satisfied by a corresponding plant. The level of each type of need is internally represented by an essential variable, ei(t) whose value is in [0; 1] and varies with time as in this equation $de_i/dt = -\alpha_n e_i(t)$. In the equation, α_n represents the decreasing rate of the essential variable. When a satisfaction level $e_i(t)$ falls bellow a given satisfaction threshold (ST), the corresponding motivation is triggered so that the agents reach a plant allowing to satisfy their needs using their cognitive maps. Thus, the satisfaction of the motivation allows an association of the current transition with the cognitive map and the node (PC) associated with this motivation. When discovering a plant, the satisfaction of the motivation associated with it is associated with the cognitive map at the location where it was found. This motivation then spreads to the graph, indicating the shortest path to reach the source from whichever location known. Thus, the agents are able to take the shortest path in order to satisfy the activated motivation. If plant from the corresponding type haven't been found by the agent,the level decreases to 0 and the agent dies. To maintain the satisfaction level of our CMAS, instead of only navigating between the three plants, it is interesting if the accumulation of in-

dividual decisions creates relevant warehouses. Warehouse creation is possible if agents are able to carry and deposit a quantity of products taken from plants. The localisation of warehouses is important because when agents reach them easily, they can increase their average satisfaction level and optimize the planning time spent looking for warehouses. In the context of situated cognition, local rules can lead to create emergent structures allowing the creation of warehouses relevant to the sorting strategy used by (Deneubourg et al., 1990). We propose generic local rules depending of the number of twithdrawals and deposits of warehouses according to the number of agents perceived. The agent can indeed, tend to favour the location which contains other agents rather than empty regions in order to create new warehouses. Thus, the perception of local agents will control the rules of withdrawals and deposits. The condition for withdrawing is computed by equation 1: the probability that an agent needs to take some plant goods increases when it perceives that the plant is less used by other agents. So, the more agents are near a plant, the more probability of withdrawal decreases and vice versa.

$$Pr_{(Taken)} = \exp^{-\lambda N_A} \qquad (1)$$

where N_A is the number of agents in the neighborhood, λ is a positive constant. Equation 2 describe the condition for deposits as follows: the probability of deposit increases with time and distance from the origin plant (when the agent is far from the origin plant where it took the last goods) and it depends of the number of the agents in the neighborhood (when the current place of the agent is frequented by other agents). For this reason, we tried to restrict the ability of agents to perceive the environment that limits the range of visual perception. Moreover, the deposit operation is also built on the concept of refueling : the agent puts goods in the warehouses that already exist.

$$Pr_{(Deposits)} = (1 - \exp^{-\alpha N_A}) * (1 - \exp^{-\beta t}) \qquad (2)$$

where α, β are environmental factors, N_A is the number of agents in the neighborhood and t is the time since the taking.

4 BUILDING OF WAREHOUSES

We use the environment as shown in figure 2a at t= 0 where the three original plants are scattered. This environment has a continuous topology and to cross it diagonally agent needs 200 time steps. The agents start to move randomly in the environment, with a limited range of visual perception that can restrict the ability of agents to perceive the environment (see

Figure2b). While passing through a plant, an agent increases its level of satisfaction and carries out the local rule of taking and transporting a quantity from the associated product. The probability of taking increases when the agent does not detect other agents next to the sources. If the decision of transporting some products is taken, the agent continues its travel. The probability of deposits increases when the agent detects other agents and it moves away from the original sources.
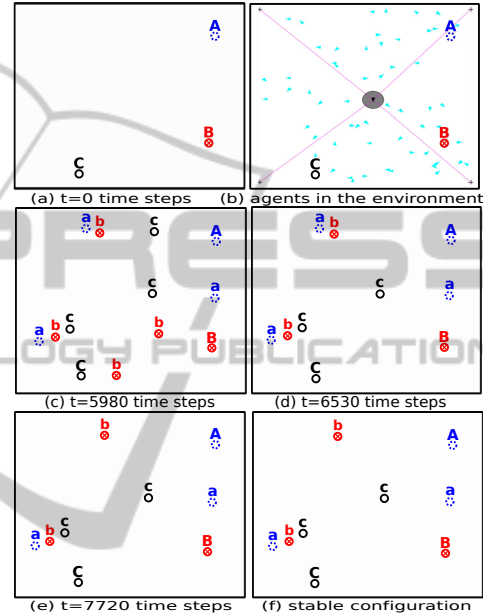

Figure 2: The figure shows 3 plants A, B and C whose positions are fixed and can deliver an unlimited amout of products. They provide warehouses of 3 different kinds ("a" is a warehouse-type A, "b" is a warehouse-type B and "c" is a warehouse-type C). Through the individual deposit process of agents, warehouses emerge and become stable after a while. The video is available in http:// perso-etis.ensea.fr/ neurocyber/ Videos/ Cognitive_Multi-Robot_System/CMAS.

This means that the locations chosen for the deposit are often common to several agents. Once deposited, goods represent a new warehouse allowing others to increase their satisfaction level. Agents also have the possibility of refueling warehouses by adding products to them. This provides stability for warehouses in relevant locations which are close to several agents in order to prevent loss. During the refuelling the available products in the warehouses will increase. However, warehouses which are abandoned or poorly visited will eventually disappear since the amount of goods available will decrease rapidly. Figures 2c and d show the disappearance of isolated warehouses. When a planning agent tries to reach a previously known warehouse and realizes that has disappeared,

two things happen: (i) the agent dissociates the current PC from the formerly-corresponding warehouse, and (ii) it resets the motivation to 0. Since the PC does not fire any more when the agent feels the need for this warehouse, there are chances that the use of transitions leading to this place be progressively forgotten. Similarly, when a new matching warehouse is discovered, the paths leading to the warehouse are immediately reinforced, making the cognitive map evolve synchronously with the environment. Finally, figure 2e shows that the CMAS converges to a stable solution with a fixed number of warehouses in fixed places at 7720 time steps and remains the same for more than 20000 time steps (see figure 2f). Here we note the emergence of stable large warehouses which in some cases emerge to create warehouses "village" (narrow regions containing all 3 warehouses). We tried to experimentally count the number of warehouses' visits compared to the number of plants' visits for 20000 time steps. The average number of visits to warehouses (115) is more important than that of the original sources (27), which shows that their location is more relevant. The same behavior also allows to keep a fixed number of warehouses (here 6) and remove others located in irrelevant places without having to use thresholds in order to limit the number of warehouses nor to specify their locations. The adaptive capability of the cognitive map, allows cognitive agents to create an emergent structure which is the stable state for the whole multi-agent system allowing the optimization of planing. Table 1 shows that agents can optimize their planning time with the help of relevant warehouses. This leads to a higher average satisfaction level. These values show the performance of our CMAS to keep the agents in their comfort zone.

Table 1: Optimization of planning time until 20000 time steps (ts).

	Only Plants	Plants+Warehouses
Planning Time	1600ts	450ts
Satisfaction Level	59.89	88.07

We calculate the average satisfaction level (ASL) according to this equation: $ASL = (\frac{1}{3nN}) * \sum_{i=1}^{n}(SL_{a_i} + SL_{b_i} + SL_{c_i})$. Where SL is the satisfaction level of warehouses, n is the number of iterations, N is the number of experiments. Based on an adaptive cognitive map, this experiment allows our CMAS (i) to optimize the planning time of agents and the number of deposits thanks to the learning of the new places of warehouses, (ii) to improve the adaptive behavior and the level of satisfaction of agents and finally (iii) to lead a fixed number of deposits in fixed places by the creation of relevant emergent structures.

5 ANALYSIS OF EMERGENT STRUCTURES

We describe here the relevance of the warehouses' location. It is a key factor of our system's performance. We placed the three original plants (A, B and C) in the summits of an isosceles triangle because we know that the center of gravity is indeed the relevant place in this case. When it includes the warehouses, it allows to optimization of the distance walked by the agents to satisfy their needs. Thus, we tried to change the satisfaction threshold (ST) to see its effect on the agents' behaviors as shown in figure 3. Basically the agents have two behaviors. The first one is the exploration mode that allows thgem to discover the environement without needing to satisfy their motivations. When the need arises and the motivation increases, the agents change the planning mode to reach a plant or warehouse to satisfy their needs. With low ST situation, the motivation decreases. However the agents continue to create relevant warehouses in the path between plants (see figure 3a and b). When ST=50, the agents can perform the two modes. In this case, we note that the agents were able to adapt to environmental change and created villages of warehouses, which consist of three types of warehouses in an appropriate location at the gravity center of triangles (see figure 3c and d). When we increase the ST (ST is stressful), the agents will seek to satisfy their needs. Figure 3e and f show that agents create villages of different kinds of warehouses which are located near to the plants, therefore the agents will stay near the villages. We also note that the change of initial position of plants doesn't change the global configuration of the environment (see the first configuration of warehouses with figure 3a,c and e; and the second configuration with figure 3b,d and f). Thus, no matter the value of ST and the position of the initial plants, the agents can create warehouses in relevant locations. Instead of browsing an Euclidean distance between the three original plants which is equal to 59.2 to satisfy the needs, agents can reduce this distance with the creation of near-perfect villages. The average Euclidean distance according to 3a is 12.08, a value well below 59.2 (the cost distance without warehouses). This shows that the simple rules of withdrawals and deposits can generate emergent structures, including the appropriateness of the location of warehouses, ensuring optimization of the planning time of our CMAS. These experiments allowed us to confirm the adaptive behavior capacity of our CMAS, based on our rules of withdrawals and deposits. Our cognitive agents are able, regardless of the position of the original plants and the shape of the environment, to choose - based
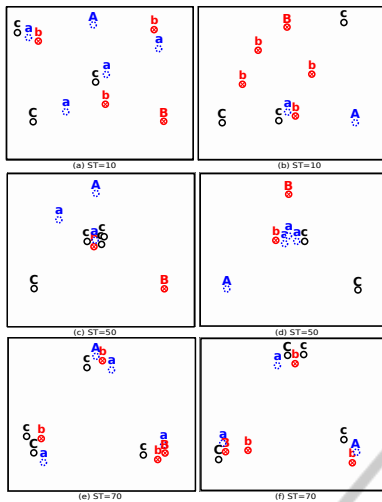
Figure 3: Different maps obtained after convergence of the CMAS according to the satisfaction threshold (ST). The agents were able to adapt to the position change by the original plants and created a village of different kinds of warehouses in relevant places which allowed them to optimize the planning time and to satisfy their needs.
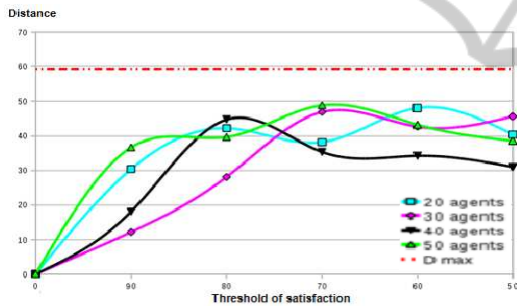

Figure 4: The average distance of the followed paths according to the ST and the number of agents. Figure shows that between warehouses (visited by agents) the average distance is much less than the distance between the original plants (D max =59.2). The figure also shows that agents are able to choose - based on an individual cognitive process - a better location for warehouses in order to optimize the planning time and satisfy their needs.

on an individual cognitive process - a better location for warehouses in order to optimize the planning time and satisfy their needs.

Now, If we focus on the paths followed by agents, figure 4 shows that between warehouses the average distance is much less than the distance between the original plants, regardless of the number of agents and the satisfaction threshold (ST). To prove this, we calculated using this equation $L_i = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} d(a,b) + d(a,c) + d(b,c)/N_1 * N_2$ the average Euclidean length L_i of the paths followed by the agents between the warehouses of different types for each simulation (duration of 20000 time steps).

Finally, we calculated the average distance $Av(L_i)$ on all of the n experiments: with L_i: distance of one experience, a,b and c: warehouses, N_1: number of agents, N_2: number of cycles. This distance reaches its maximum 48.86 which is less than 59.2 (distance between the original plants) with 50 agents and with TS equal to 70. This shows that the warehouses allow the agents to optimize their paths in an emergent way, to satisfy their needs without the use of an optimization model of navigation. Thus, the simple rules of withdrawals and deposits can generate emergent behaviors, including the appropriateness of the location of warehouses, ensuring the optimization of the planning time of our CMAS.

6 EVALUATION OF EMERGENT WAREHOUSES' LOCATION

To make the application field of our CMAS more concrete and to test the adaptability of our agents when changing the context, we have added customers to the environment in order to switch back to a classical warehouse location problem: We distribute a set of products to a set of demand zones either directly from production plants or through warehouses. Indeed, trying to use MAS to supply chain-like problems is not new (Yadati et al., 2010); however in most cases dedicated and pecialized agents are used. In this part we describe a formal instrument of performance that shows how our CMAS can be adapt to an optimization problem and reasonably give a sound result related to MAS_{IAA} (Laroque et al., 2010) which is a deposit system that takes inspiration from ant sorting skills used to build a cemetery and with a MAS_{Random} which makes random deposits. Initially, the warehouses which have a limited capacity don't exist. We assume that the transport cost is lower between plants and warehouses than between plants and customers. A set of potential sites to locate the warehouses is determined *a priori* (in our case a regular paving of the environment to try to find the optimal set of locations). We assume that the production plant capacity is sufficient to satisfy the customer's requests. We want to determine which warehouses should be used and where. To determine then the product flows (i) between the production plants and the customers (direct delivery), (ii) between the production plants and the warehouses (internal traffic) and (iii) between the warehouses and the customers (indirect delivery). To materialize the cost of transport in our simulations, during a deposit, the agent must wait for $\theta = 29$ second which represents the time of the creation of such deposit, so the level of needs will decrease.

273

6.1 Modeling the Warehouse Location

We start to express the warehouse location problem and its constraints, and then we designed a mixed integer program. We used the following sets of indexes in our model: Product (P), Demand zone (customers) (D), Production plant (U) and Warehouse (W). To express the model, the following **data** are required: d_{up} is the number of units u of product p required by demand zone d. c_w is the maximum number of units u of products which can come through the warehouse w. $dc_{pud}, ic_{puw}, ic'_{pwd}$ are the unit delivery cost of product p from plant u and warehouse w to respectively demand zone d, warehouse w and demand zone d. wc_w is the fixed cost associated to the use of warehouse w. **The decision variables** are W_w: Binary variable indicating the use of w (use of w =1; else= 0). $Qd_{pud}, Qiw_{puw}, Qid_{pwd}$ are the number of units p shipped by plant u and warehouse w to respectively demand zone d, warehouse w and demand zone d. As **objective function**, we chose the minimization of the total cost of transport to converge to an optimal solution.

$\min(f) = \sum_{w=1}^{W} wc_w.W_w + \sum_{p=1}^{P} \sum_{u=1}^{U} \sum_{d=1}^{D} dc_{pud}.Qd_{pud}$
$+ \sum_{p=1}^{P} \sum_{u=1}^{U} \sum_{w=1}^{W} ic_{puw}.Qiw_{puw}$
$+ \sum_{p=1}^{P} \sum_{w=1}^{W} \sum_{d=1}^{D} ic'_{pwd}.Qid_{pwd}$

where the **constraints**: The model must satisfy the demand zone from warehouses or/and plants.

$\sum_{u=1}^{U} Qd_{pud} + \sum_{w=1}^{W} Qid_{pwd} = d_{pd} (\forall p \in P \forall d \in D)$,

The model must not exceed the warehouses capacity.

$\sum_{p=1}^{P} \sum_{u=1}^{U} Qiw_{puw} \leq c_w.W_w (\forall w \in W)$

The quantities must be concerned between plant to warehouses and warehouses to demand zone.

$\sum_{u=1}^{U} Qiw_{puw} = \sum_{d=1}^{D} Qid_{pwd} (\forall p \in P \forall w \in W)$

The model must also respect the constraints of integrality and non negativity and ensure flows between plants, warehouses and demand zone.

$W_w \in 0,1, Qd_{pud} \geqslant 0, Qiw_{puw} \geqslant 0, Qid_{pwd} \geqslant$
$0, (\forall w \in W, p \in P, u \in U, d \in D,)$

6.2 MAS vs. Linear Programming

To make the comparison between the LP and the MAS approachs possible, we must extract from each MAS system (MAS_{Random}, MAS_{IAA} and our CMAS) the customer demand to inject it into the LP in order to have an equivalent demand (because it's not the same in the three systems). Thus the LP will give us the optimum warehouses' positions for each system. For this reason, our instrument of performance consists of the computing of the average distance between MAS

warehouses' location and LP warehouses' location for each system. Unlike LP, the MAS is not deterministic because it does not provide the same results on several tests with the same parameters. We performed 10 tests of MAS and 1 test of linear programming and then compared the average of the 10 tests of MAS with LP. For each warehouse W_i of LP (equation 3), we summed the minimum distance between the W_i warehouse and the corresponding MAS W_i':

$$d_i = \min(d(W_i, W'_j)) \qquad (3)$$

where d is the Euclidean distance on the environment. The average distance of an experiment Exp_k (equation 4) is summed with a penalty p which depends on the difference between the number n of warehouses in LP and the number n' of warehouses in MAS:

$$d(Exp) = ((\sum_{i=1}^{n} \frac{d_i}{n}) + p(|n' - n|)) \qquad (4)$$

Finally, we calculated the average distance of all the Ne experiments. Table 2 shows that our CMAS could optimize the average cost of the warehouses location better than the MAS_{IAA} and the random system MAS_{Random}. We also note that with a homogeneous distribution of plants and customers, we obtain a small average distance with a quasi-optimal number of deposits compared to mathematical programming with a very low standard deviation, which means that in CMAS the warehouses are well placed if we compare them to a MAS_{IAA} and to a MAS_{Random} and close to the locations given by the mathematical programming solution in the environment. The results obtained confirm the performance of the emergent structures based on cognitive process which allows us to have good solutions that approximate the solution of LP in a time equal to 40 minutes (non optimized java simulation), while LP approximately requires on average several hours with a very large standard deviation, going from 2 minutes to more than 24 hours (GLPK software) to solve the same problem (with the same configuration of machines).

Table 2: The average distance of the MAS systems relative to the LP solution.

	MAS_{Random}	MAS_{IAA}	CMAS
Average number of deposits	22	15	4
Average cost of the solution	40671.3	55606.4	15210.6
Average distance relative to LP	42.7	27.3	6.9
Standard deviation of average distance	9.0	10.7	1.8

7 CONCLUSIONS

In this paper we describe how the local learning with the accumulation of individual decisions advocates the creation of new emergent structure. We suggest a set of simulations that analyze the positive impact of individual behavior in the improvement of the global performance of the system. This individual behavior that is based on the bio-inspired cognitive map, allows the agents to avoid planning problems before a complete exploration of the environment. It also leads to the same results as a pheromon-based system without the need to leave a physical trace in the environment. Thus, the coupling of the individual behavior with the embodiement of the agents (satisfaction of their motivations) can solve multi-objective planning problems although formally the algorithm is not able to "mix and to merge and to optimize" several objectives. In conclusion, we suggest an emergent multi-objective optimization. Finally we suggest the evaluation of the emergent structures in MAS by comparing our CMAS, the MAS based on cemetery organization of ants and the random MAS with a linear programming approach. The results obtained confirm the performance of our emergent behavior based on cognitive processes which allow us to have adequate solutions that approximate the linear programming solution. To study the limits of emergent structures in real world, we started to validate the adaptive capability of the cognitive map in a real multi-robot system (Chatty et al., 2012) and now we are trying to add the deposit system in the multi-robot system.

ACKNOWLEDGEMENTS

REFERENCES

Billing, E. A. (2010). Cognitive perspectcoives on robot behavior. In *ICAART (2)*, pages 373–382.

Bonabeau, E. and Theraulaz, G. (1994). *Intelligence Collective*. Hermes.

Brooks, R. A. (1994). Coherent behavior from many adaptive processes. In *in SAB*, pages 22–29, Cambridge, MA, USA. MIT Press.

Chatty, A., Gaussier, P., Kallel, I., Laroque, P., and Alimi, A. (2012). Adaptation capability of cognitive map im-

proves behaviors of social robotics. In *in IEEE ICDL-EpiRob (to appear)*.

Chatty, A., Kallel, I., Gaussier, P., and Alimi, A. (2011). Emergent complex behaviors from swarm robotic systems by local rules. In *IEEE Symposium Series on Computational Intelligence, (RiiSS)*, pages 69 –76.

Deneubourg, J. L., Goss, S., Franks, N., Franks, A. S., Detrain, C., and Chrétien, L. (1990). The dynamics of collective sorting robot-like ants and ant-like robots. In *SAB*, pages 356–363, Cambridge, MA, USA. MIT Press.

Gaussier, P., Revel, A., Banquet, J. P., and Babeau, V. (2002). From view cells and place cells to cognitive map learning: processing stages of the hippocampal system. *Biological Cybernetics*, 86(1):15–28.

Gaussier, P. and Zrehen, S. (1994). Avoiding the world model trap: An acting robot does not need to be so smart! *Robotics and Computer-Integrated Manufacturing*, 11(4):279 – 286.

Laroque, P., Pirard, F., Archambault, F., Quoy, M., and Gaussier, P. (2010). Comparing multi-agent system and mixed-integer programming approaches on a simple optimization problem. In *IC-AI*, pages 977–981.

Lubenov, E. V. and Siapas, A. G. (2009). Hippocampal theta oscillations are travelling waves. volume 459, pages 534–539. Nature Publishing Group.

Martinet, L.-E., Sheynikhovich, D., Benchenane, K., and Arleo, A. (2011). Spatial Learning and Action Planning in a Prefrontal Cortical Network Model. *PLoS Comput Biol*, 7(5).

Mataric, M.-J. (1992). Designing Emergent Behaviors: From Local Interactions to Collective Intelligence. In Meyer, J.A., R. H. and Wilson, S., editors, *Proceedings of the Second Conference on Simulation of Adaptive Behavior*, pages 1–6. MIT Press.

Mathieu, P. and Secq, Y. (2012). Environment updating and agent scheduling policies in agent-based simulators. In *ICAART (2)*, pages 170–175.

Matsui, T. and Matsuo, H. (2012). Analysis for distributed cooperation based on linear programming method. In *ICAART (2)*, pages 228–233.

O'Keefe, J. and Nadel, L. (1978). *The hippocampus as a cognitive map / John O'Keefe and Lynn Nadel*. Clarendon Press ; Oxford University Press, Oxford.

Pepin, N., Simonin, O., and Charpillet, F. (2009). Intelligent Tiles: Putting Situated Multi-Agents Models in Real World. In AAAI, A., editor, *ICAART*, Porto, Portugal.

Portugal, D. and Rocha, R. P. (2012). Extracting topological information from grid maps for robot navigation. In *ICAART (1)*, pages 137–143.

Prodan, I., Olaru, S., Stoica, C., and Niculescu, S.-I. (2012). Predictive control for trajectory tracking and decentralized navigation of multi-agent formations. In Filipe, J. and Fred, A. L. N., editors, *ICAART (2)*, pages 209–214. SciTePress.

Simonin, O., Ferber, J., and Decugis, V. (1998). Performances analysis in collective systems. ICMAS, pages 469–. IEEE Computer Society.

Yadati, C., Witteveen, C., and Zhang, Y. (2010). Coordinating agents - an analysis of coordination in supply-chain management tasks. In *ICAART (2)*, pages 218–223.