Acta Universitatis Apulensis,
ISSN 1582-5329, 2011,
Special Issue on Understanding Complex Systems,
Eds. Barna Iantovics, Ladislav Hluchý and Roumen Kountchev

APPLICATIVE ISSUES FROM SELF-ORGANIZED SYSTEMS BASED ON SOCIAL INSECTS MODELING - A TUTORIAL -

Cyrille BERTELLE, Antoine DUTOT and Rawan GHNEMAT

Abstract. Self-organization is common in natural systems. This tutorial describes some of these systems, specifically from insect societies like in bees, termites or ant colonies. In a first part, a modeling process is explained. Objects and phenomena targeted by these methods are presented. Natural or social complex systems are the context of these objects and phenomena. Basic algorithms presented for example in [10] are given. These algorithms belongs to the class of swarm intelligence methods describing how a network of interacting entities can lead to emergent properties of the whole system. In a second part, more original applications are presented, based on extensions of these basic algorithms in order to model ecosystems, urban dynamics or to propose a decentralized method to distribute simulations over dynamical communication graphs.

Keywords: *swarm intelligence, complex systems, ant colony optimization, self-organization, emergent properties*

2000 *Mathematics Subject Classification*: 68Txx.

1. Complexity and Self-Organization

Physic, Biology, Computer Science or Human Science give many examples of systems where the global behavior is the result of interactions between homogeneous or heterogeneous entities. The figure 1 shows such emergent behavior from collective movement of fish banks or bacteria colonies.

Figure 1: Natural self-organization in natural systems - fish banks and bacteria colonies (from Tel-Aviv university - prof. Eshel Ben-Jacob laboratory [5])

Some elements to understand such self-organizations can be deduced from these observations in natural systems: (i) the entity interactions are essential in order to generate the system formation, (ii) there is two levels of properties involved in such phenomena, local and global ones, (iii) the entity interactions are expressed within a network in continuous evolution.

Our purpose is to model and to analyze social organizations inspired from these natural systems. Moreover, we want to highlight the essential role of space and so, how spatial interactions are the catalyst of emergent properties which appear in the formation of these complex systems.

On Figure 2, we describe a two-level model of spatial self-organizations with interactions in both directions between these two levels: the emergence of organizations from entities interactions but also the feed-back process describing how organizations are regulating their own entities.

In the following, we focuss our study on how building concrete algorithms to implement such phenomena involving adaptive complex systems. These algorithms are mostly belonging to swarm intelligence methods.

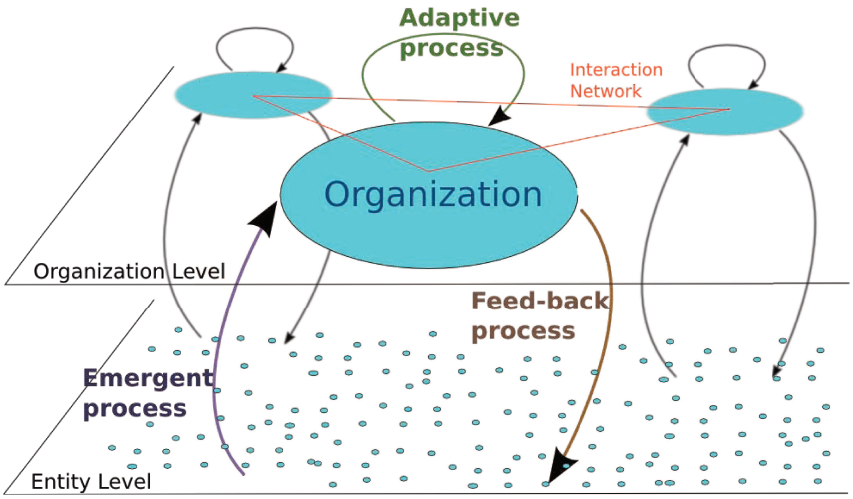


Figure 2: Complex spatial organizational model

2. Swarm Intelligence

Swarm intelligence algorithms [8, 10, 21] find roots in various aspects of computer science development: from the evolution of the network architectures that have led to distributed computing, but also from the evolution of software engineering that have led to distributed artificial intelligence paradigm.

Distributed programming development starts from the evolution of the sequential architectures toward parallel ones. During many years, the major concept in parallel computing has been the master/slaves model. This model is a centralized decomposition of tasks, suitable for high performance super calculator. The core of such computer is a CPU grid which is a predefined and static topological network of connection between the CPU. With the increase of the speed of communication between computer networks, distributed computing finds efficient solutions in using huge computer networks. Such networks have dynamical aspect such as disconnection of one of the computer and so decentralized approaches are now indispensable.

Moreover, software engineering development has seen a fundamental change in the transition from procedural programming to object programming. Some of the major aspects of this evolution were the abstraction on method accesibility (public or private) and the decomposition of program in structural units interacting together. Agent-based programming becomes a natural evolution of this software engineering. The program units increase their "autonomy", being able to implement for themselves a life cycle or a data knowledge.

Swarm Intelligence is based on such distributed solvers expected to interact together. In this approach, there is no plannication of any kind of master solvers, splitting a problem in smaller ones and then assemblying the subsolutions of each slaves. In swarm intelligence algorithms, the computation is decentralized and the global solution emerges from these distributed solver interaction.

The most famous swarm intelligence methods are Particle swarm optimization (PSO) [20] and ant systems [10]. In PSO, the distributed solvers are moving virtual particles. These virtual particles are collectively moving on a solution space. The collective displacement of the particles system is inspired from C. Reynolds Boids [29] system which aims to simulate the collective behavior of fish banks or bird flocks. The virtual particles of PSO interact by exchanging information in order to discover the best solution over the space solution.

In ant Systems the distributed solvers are ants moving in an environment. The emerging solutions are achieved through a concept called stigmergy. This concept is based on a system of indirect communication whose support is the environment. It is inspired by the behavior of natural ants depositing pheromone trails on the ground to communicate with their peers.

3. Ant Systems

3.1 Self-organization in natural ant systems

Natural ant behaviors have fascinated complex systems researchers since many years by their capacities to organize themselves on various aspects. On the figure 3, they collaborate to manage a broken way, making a living bridge.

Computer Science researchers are using ant colonies behavior to design bio-inspired methods and algorithms. Their specific reactive individual behaviors are suitable to design efficient cooperation mechanisms in order to implement automatic processus leading to self-organization formation.

To achieve this objective, only the few essential aspects of the ant behavior which lead to the self-organization processes, have to be understood and then described with mathematical formulations.

Engineering applications of such artificial ant colonies systems start to be developed nowadays, specifically for problems which can be described in term of graph optimization, task allocation or clustering.

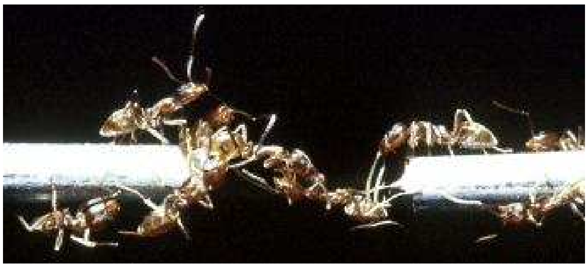


Figure 3: Self-organization within natural ant colonies (from [12])

On figure 4, two examples of self-organization in natural ants are presented. On the left side, the well-known Deneubourg experiment consists to highlight with a very simple device the ant foraging problem. The ant objectives is to find the optimal way from nest to food source, using pheromone trail deposition. On the right side, cemetery clustering formation are shown at 4 successive times: ants form piles of corpses to clean their nests. Each of them has elementary actions, unknowing the whole situation, but dealing only with local information. There is no supervisor to lead the piles formation which emerges from ant interactions.

3.2 Ant Colony Optimization

Ant colony optimization [10] has been originally proposed to solve the famous Traveling Salesman Problem (TSP). This problem consists in finding the shortest cycle which links N interconnected towns within weighted graph with only one pass in each town. The problem is described by a graph where nodes C_i are towns and weighted edges D_{ij} are the distance between towns. $T_{ij}(t)$ are pheromone quantity deposited by ants on the edge (i, j).

When an ant is on the town i, it computes the probability to go to the non yet visited town j by the formula:

$$P_{ij}^k(t) = \begin{cases} \dfrac{(T_{ij}(t))^\alpha \left(\dfrac{1}{D_{ij}}\right)^\beta}{\sum\limits_{l \in J_k(t)} (T_{il}(t))^\alpha \left(\dfrac{1}{D_{il}}\right)^\beta} & if \quad j \in J_k(t) \\ 0 & if \quad j \notin J_k(t) \end{cases}$$

where

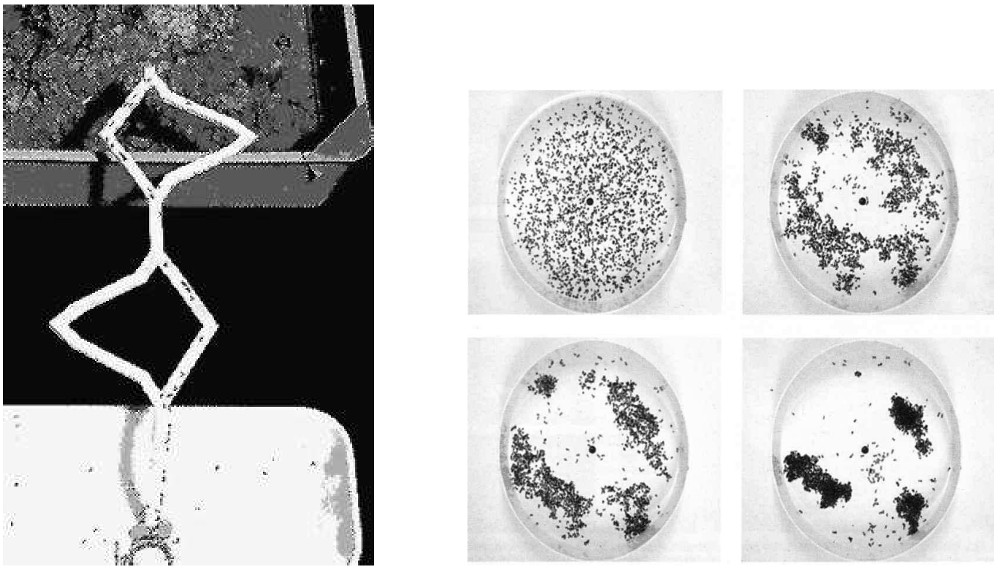


Figure 4: Optimization in natural ants collective behavior: foraging and clustering (from [10])

- J_k is the set of towns not yet visited by the ant k;
- α and β allow to control the relative importance of the 2 quantities controling the ant behavior: the pheromone trail quantity (T_{ij}) and the distance (D_{ij}).

When an ant has found a solution as a good cycle between towns, it deposits some pheromone on all of the edges of the cycle, inversely proportional to the length of the cycle (L_k):

$$\delta T_{ij}^{k}(t) = \begin{cases} \frac{Q}{L_k} & if \quad (i,j) \;\; is\ a\ edge\ of\ the\ cycle \\ 0 & elsewhere \end{cases}$$

Where Q is a constant parameter

On each edge (i, j), the pheromone quantity is updated from step t to step t+1, by adding all the contributions of each ant to previous pheromone quantity:

$$T_{ij}(t+1) = \rho T_{ij}(t) + \sum_{k=1}^{m} \delta T_{ij}^{k}(t)$$

Where ρ is an evaporation factor which allow that some first path/solution can be replaced by better ones.

3.3 Ant Clustering

Ants and other social insects deal with cooperative way in distributed clustering, like cimetery. From the beginning of aggregations of objects, small clusters formation appear and act themselves on social insects by stigmergy. This mechanism make the clusters formation increase with time.

The algorithm consists in making a great number of autonomous ants move in random direction inside an environment of objects to agregate them in clusters.

At each step, an ant is in one of the 3 situations:

1 The ant moves without carrying anything and meet no object. In this case, the ant continues to move randomly

2 The ant moves without carrying and meet an object. The ant can take this object to carry it. The probability of the ant take the object is:

$$P_p = \left(\frac{k_1}{k_1 + f} \right)^2 \tag{1}$$

- f is a value corresponding of the number of objects perceived in the ant neighborhood
- k_1 is a threshold, making this probability between 0 and 1, according to its relative position with k_1.

3 The ant moves and carries an object. The ant can leave this object on the ground. The probability of the ant to leave the carried object is:

$$P_d = \left(\frac{f}{k_2 + f} \right)^2 \tag{2}$$

- f is a value corresponding of the number of objects perceived in the ant neighborhood
- k_2 is another threshold, making this probability between 0 and 1, according to its relative position with k_2.

4. Applications

4.1 Methodology for applicative issues development from bio-inspired algorithms

Original research works, from the research team RI2C (French acronym for swarm intelligence and interaction networks) of the laboratory LITIS, is developed in this section. They belongs to a specific methodology of development to applicative issues about living and social complex systems, within distributed computing. Two major goals are defined and consist in (i) understanding and modelling the organization and self-organization of living systems using appropriate models and simulations, (2) taking inspiration from living systems to design new methodologies, new models and new algorithms suited to distributed computing.

This methodology concerns applicative developments on (i) modelling and simulating aquatic ecosystems (fluid flow and trophic chains organizations in multi-scale and multi-model approaches), (ii) implementing dynamical distribution of multi-agents simulation over computer network, dealing with load balancing, (iii) managing services execution and spreading over distributed systems composed of mobile resources, (iv) modeling urban dynamics.

In order to achieve these applications development, we propose a general methodology which can be summarized with the figure 5. In this figure, we refere to the main problem of our research group which consists in detecting structures or organizations inside complex systems which can be the model of various applicatives domains as mentionned previously. Our objectives is to integrate inside our simulations, some models of feed-back control of these emergent organizations over their own constituents. Swarm intelligence algorithms are the main basis of our implementation.

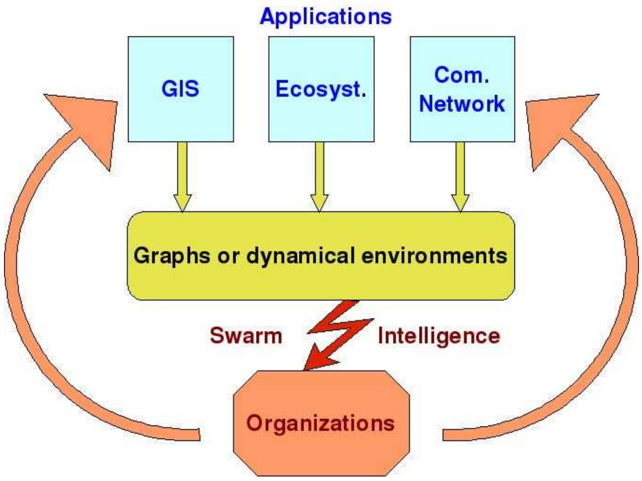


Figure 5: Conceptual modeling process for applicative complex systems

In the figures 6, 7, 8, we show an application modeling virtual ecosystem composed of dynamical organizations of some species, represented here by specific color. Boids models are generalized to colored boids behavior where some attraction rules between species of the same color are introduced while repulsion rules between species of different color are defined. Interaction between these emergent organizations are shown in these figures.

4.2 Structure detections in distributed simulations

This research actions deal with dynamical agent-based simulation distribution over computer network based on ant system. The goal is to distribute the previous simulation interacting entities on a computer network: the communication has to be minimized, placing highly communicating entities on the same computer. Load balancing must be also considered in order to efficiently distribute entities to all the computers, respecting their power capabilities.

The proposed solution consits in developing an innovative algorithm called AntCO2, acronym for Ant Competing Colonies [16].

AntCO2 principle is an extension of the classical ant system with dynamical colored graph and colored pheromons. Several ant colonies, each of a distinct color travel through the dynamical graph (as described in the figure 9). The graph nodes represent the entities. The graph edges represent the interactions between the entities and are weighted by the importance of the interaction

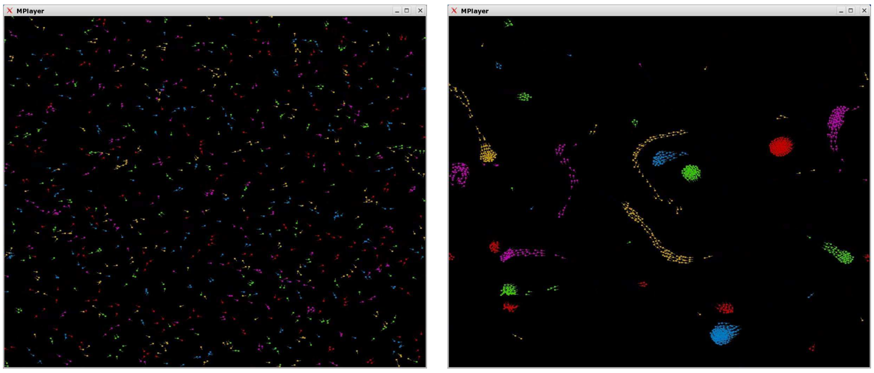


Figure 6: Ecosystem modeling based on colored boids simulation: the left figure describe the initial situation and the right one describe the simulation after several steps, showing the formation of dynamic organizations

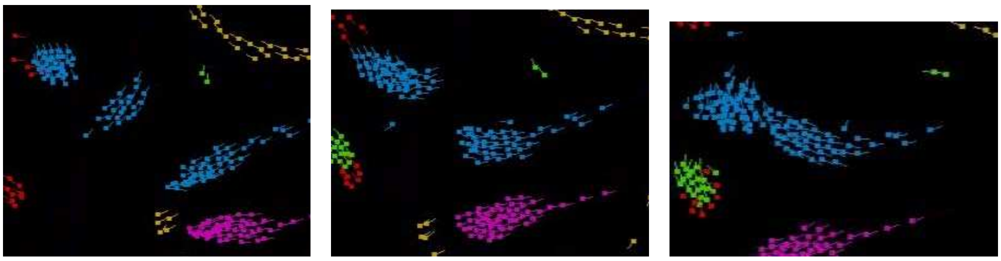


Figure 7: Emergent organizations during ecosystem simulation: interaction of elements of the same category leads to fusion and agregation

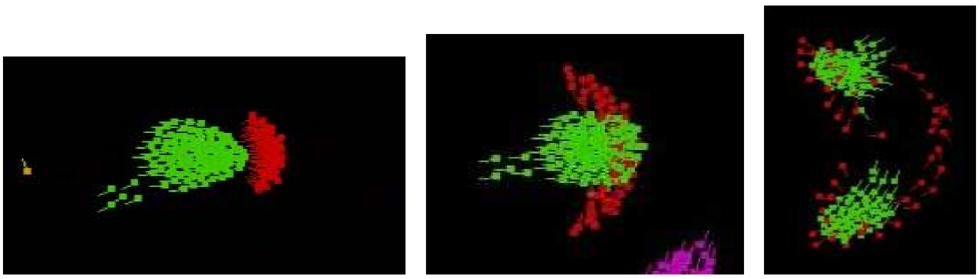


Figure 8: Organization evolution during ecosystem simulation: interaction of structures of different categories which are broken by the interaction processing

between these entities. The graph evolves with time, as the simulation evolves. If an entity appears or disappears, a node appears or disappears and the same for interactions and edges. If an interaction increase or decrease, the value on the edge similarly increase or decrease.

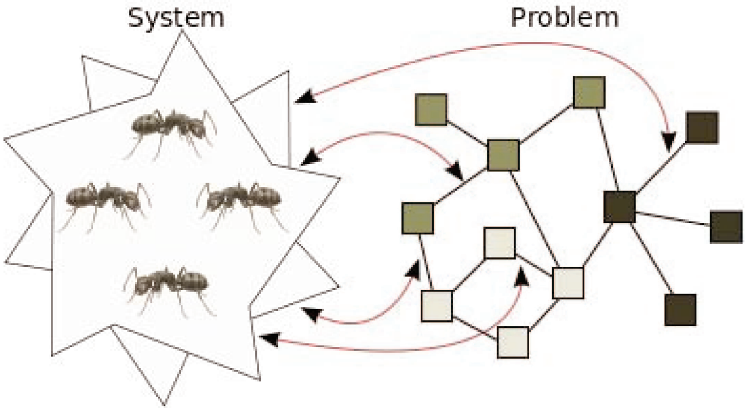


Figure 9: The general model: The system evolves on the graph of the problem, modifies it and is influenced by its own modifications. Finally solutions are directly observed on the graph (from [2]).

Ants of a colony drop pheromons of the color associated to the colony. The ants of a colony are attracted by the pheromons of their own color, and repulsed by the pheromons of other colonies. All ants are attracted by edges of the graph representing high interactions between entities.

The originality of this approach lies in the fact ants collaborate as in other algorithms inside a colony, but also enter in competition with other colonies. This collaboration versus competition mechanism allows to find an equilibrium where colonies maintain areas in the dynamic graph corresponding to clusters of highly interacting entities.

An area of the graph is conquered by a colony when it is covered by a dominant pheromon color. As pheromons evaporate continuously, the areas must be maintained by ants that constantly pass and stay in the area because they are attracted by their own pheromon and highly interacting edges. If the dynamical graph evolves, a cluster that disappears will be forgotten by evaporation and ants will try to conquer other areas. Therefore the algorithm solutions evolve with the problem. This adaptivity of the algorithm allows to perform structure detection on evolving problems.

Colonies are associated with computing resources. When an area is colored by a colony the entities of this area are run by the associated computing resource. The number of ants of a colony allows to distribute the simulation on computing resources of different powers. Small colonies will tend to conquer smaller areas since they can maintain their colored pheromons on less nodes of the graph at a time. Such colonies will be associated with computing resource of small power. Conversely large colonies will conquer one or several zones of the graph of larger importance for more powerful computing resources.

We measure the quality of the distribution using two criteria named r1 and r2. The first criterion measure the ratio between the number of effective communications and the total communications. In this criterion, each interaction is a communication between two entities. When entities are on distinct computing resources we say that the communication is effective. The second criterion measure the ratio between the minimum load of a computing resource and the maximum. These loads are normalized since all computing resources are not of equivalent power. The two criteria are contradictory and therefore a solution to the problem is a trade-off between the two.

One of the important advantage of this algorithm is that it does not use a centralized approach. Ants do not need any global control and therefore it is quite easy to distribute, in order to accommodate large simulations.

AntCO2 results show the capability of this algorithm for cooperation and competition process leading to emergent clustering. On figure 10, graph clusterings are obtained by this algorithm. Its major characteristic is its decentralized process which confer flexibility and robustness properties.

The ideas in AntCO2 can be used to create an organization or community detection algorithm only, without the simulation distribution aspects. This has been done as shown on figure 11 were AntCO2 is used to detect communities in a graph created using the Amazon.com site. As before, the communities are shown using colors. The graph is as follows : for each book listed on the site a node is created in the graph. For the books on the site, five books are given as also bought by clients that bought this item. Therefore for these five books we create five new nodes and we add edges between them and the initial book. We then restart this algorithm for each five new books, etc., until a given depth. This creates a graph where books on similar topics appear in clusters of highly connected nodes, and with fewer edges between the clusters. This technique works well on static graph, but is still in work for dynamic graphs.

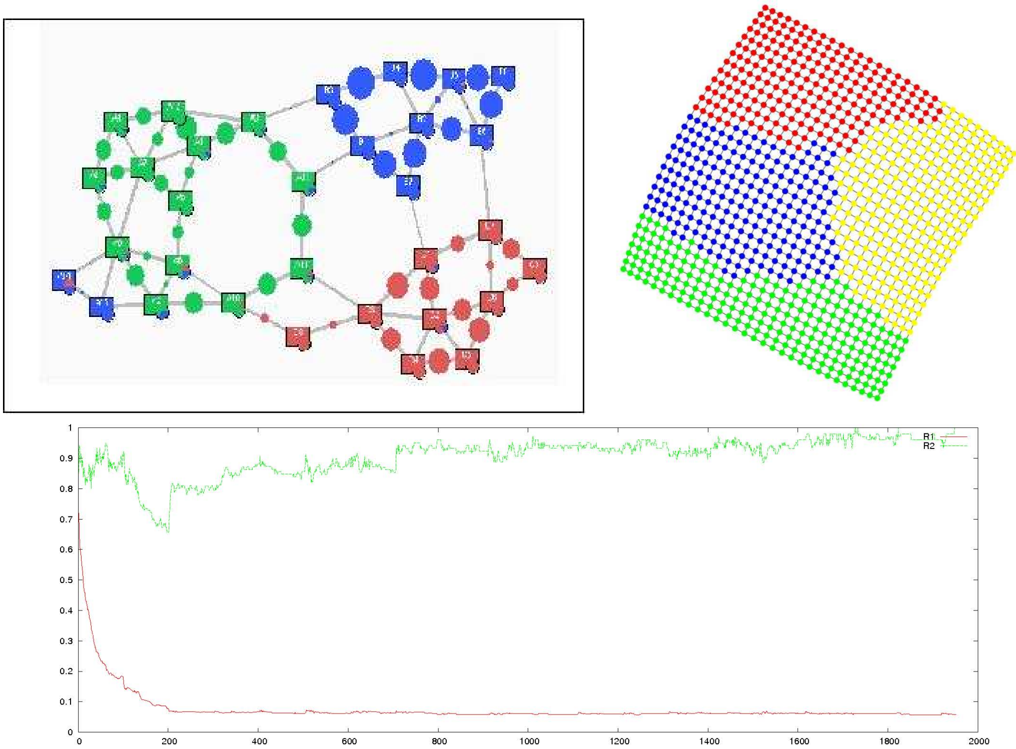


Figure 10: AntCo2 algorithm for graph clustering: on the top left, the output of the computation on a communication network; on the top right, the output on a regular grid; on the bottom, typical curves describing the evolution of the criteria r1 and r2, measuring the distribution efficience and the load balancing efficience

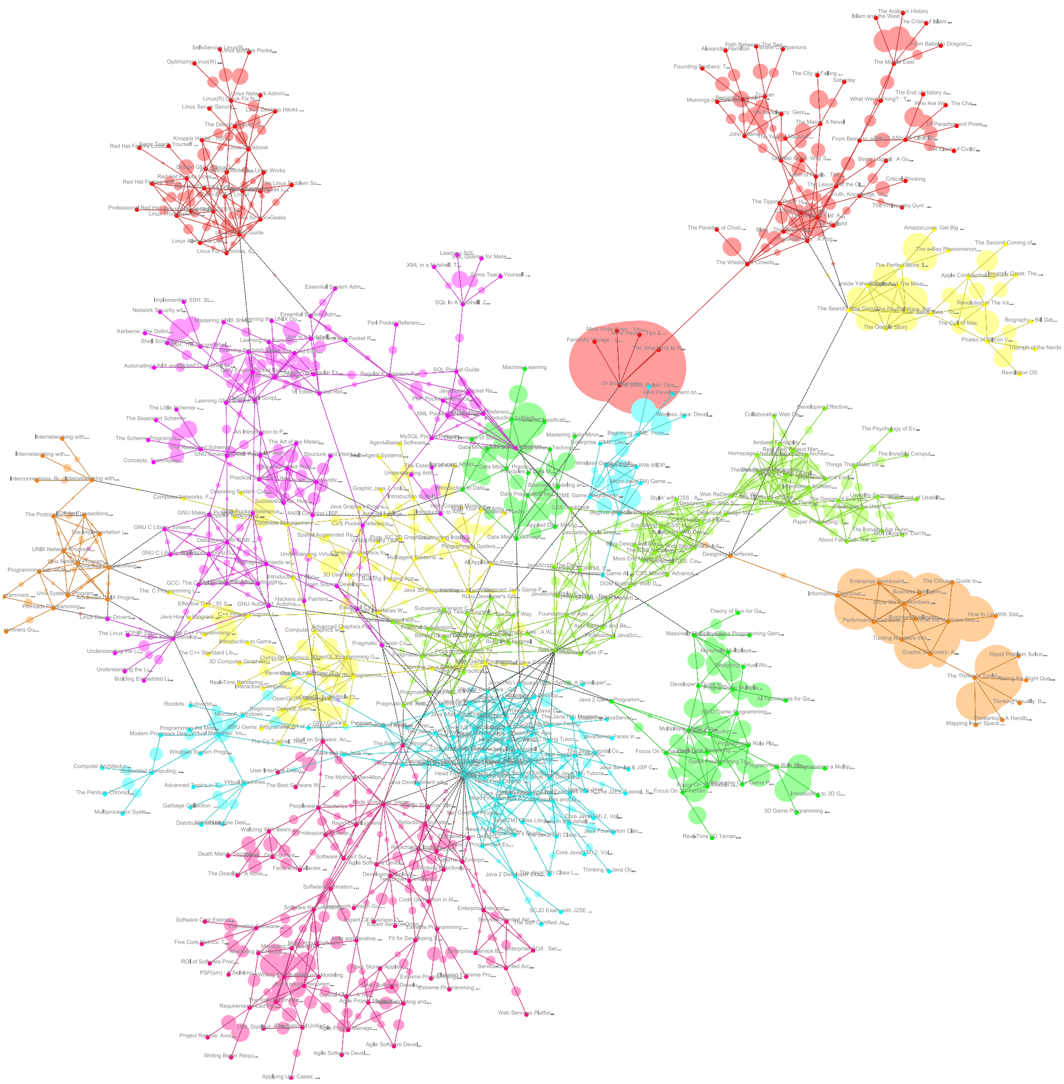


Figure 11: Communities detection using AntCo2 over Amazon graph

4.3 Urban dynamics simulation

Social and human developments are typical complex systems. Urban development and dynamics are the perfect illustration of systems where spatial emergence, self-organization and structural interaction between the system and its components occur [3, 4, 6, 7]. In figure 12, we concentrate on the emergence of organizational systems from geographical systems. The continuous dynamic development of the organization feed-back on the geographical system which contains the organization components and their environment. To analyse or simulate urban dynamics, nowadays, we can use the great amount of geographical databases directly available for computational treatment within Geographical Information Systems [27, 30]. On the organizational level description, the new development of multiagent systems (MAS) allows nowadays to develop suitable models and efficient simulations [13].

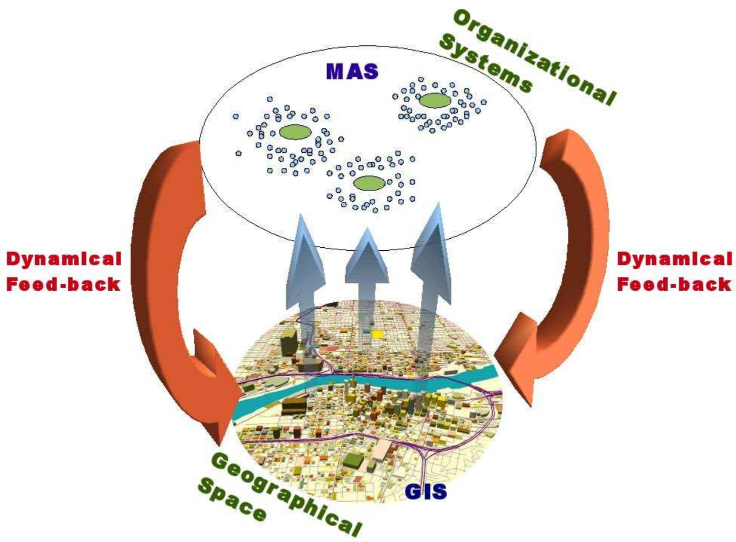


Figure 12: Complexity of geographical space with respect of emergent organizations

The applications we focus on in the models that we will propose in the following, concerns specifically the multi-center (or multi-organizational) phenomona inside urban development. As an artificial ecosystem, the city development has to deal with many challenges, specifically for sustainable development, mixing economical, social and environmental aspects. The decentralized methodology proposed in the following allows to deal with multi-criteria problems, leading to propose a decision making assistance, based on simulation analysis.

Gentrification phenomena can be modelled using such methodology. It is typically a multi-criteria self-organization process where appears emergent coming of new population inside urban or territorial areas. This new population firstly attracted by some criteria, brings some other charateristics which are able to modify and to feedback over the environment. Cultural urban dynamics processes in urban areas are also such complex systems where multi-criteria must be taken into account.

The ant clustering shows some spatial self-organizations but has the specificity to generate clusters at random places. According to the first random moves that the ants start to do in the beginning of the algorithm, some material will initiate aggregation and the clustering processus will complete this aggregation from these initial random first aggregations. To simulate some urban dynamics, we need to introduce specific location with respect to city center, for example. The clustering here will represent the people usage of these centers or equipments and we need to introduce an attractive effect by using a pheromon template. This method follows the algorithm known as Ant Nest Building [14]. In ant colonies, the center corresponds to the position of the queen which needs to build the nest and the ant colony moves around it to protect the nest by various material taken on the ground. The queen emits a pheromon which allows to attract the ants during their building. The ant has to deposit the material carried only if the pheromon quantity perceived belongs to a specific range. We use an attractive fonction called P_t, corresponding to a pheromon template and represented by the top part of the figure 13.

Using this template function, we remplace in the clustering algorithm, the two previous probabilities defined previously in equation (1) and equation (2) by

$$P'_p = P_p(1 - P_t) \tag{3}$$

$$P'_d = P_d P_t \tag{4}$$

On figure 13, we can see a single queen simulation. It is possible also for each queen to emit many different kinds of pheromons : we called them colored pheromons [17]. Each colored pheromon will attract only the ants associated to its color. On figures (b) and (c), simulations on RePast [15, 22, 28] are provided at successive times. The last figure shows the adaptive mechanism of the queen which grows with time according to the material density around it, like in natural observations.

On the figure 14, we present a complete simulation with several centers and several queens [17]. On each center, a queen is emitting several colored pheromons and are able to attract some multi-colored material according to their initial location. Simulation outputs at different iteration times are presented, from RePast implementation and OpenMap GIS visualization.

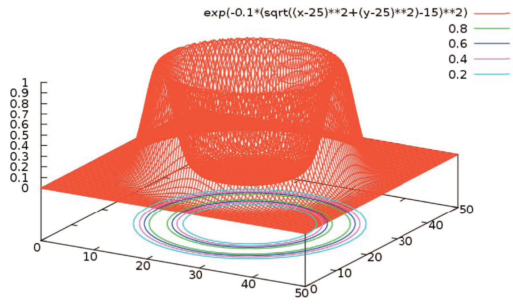
The multi-template modelling can be used to model cultural equipment dynamics as described in figure 15. On this figure, we associate a queen to each cultural center (cinema, theatre, ...). Each queen will emit many pheromon templates, each template is associated to a specific criterium (according to age, sex, ...). Initially, we put the material in the residential place. Each material has some characteristics, corresponding to the people living in this residential area. The simulation shows the self-organization processus as the result of the set of the attractive effect of all the centers and all the templates.

4.4 Possible congestion areas detection

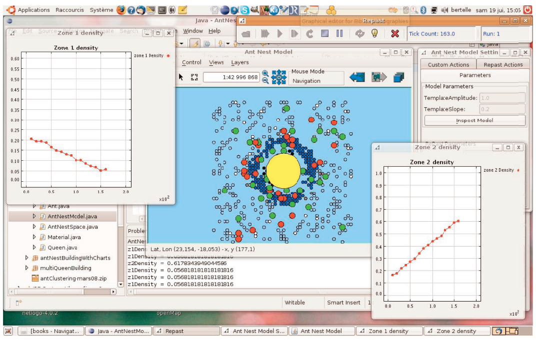
In this work we try to detect roads where a possible congestion might appear using only the road network structures. The goal of this work is not to provide a traffic simulation, but to collect information on the structure of the road network. We do this in order to gain knowledge on which parts of the network can become problematic for traffic flow, notably in case of emergency evacuation after an accident for example.

Using a GIS, we extract a graph from the road network. Each road becomes an edge and each road intersection becomes a node. Edges are extended with capacity information allowing to know approximately the maximum number of vehicles that can transit on the corresponding road at a given time.

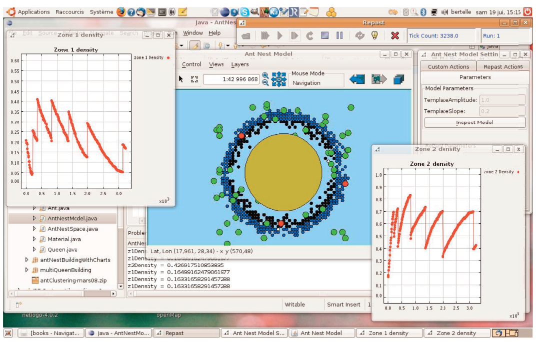
We use simple ant-like entities that represent vehicles traveling on the graph. At each time there is always the same number of ants on the graph. This number is chosen empirically. Actually as many ants as there are nodes in the graph. At start each ant begins its travel on a node chosen at random. Then the simulation is iterative. At the first step, ants can choose an edge of the node they are on at random. Once the edge is chosen, they enter a



(a) template function



(b) simulation on Repast: after few step



(c) simulation on Repast: after queen adaptive development

Figure 13: Adaptive queen behavior modelling: according to its surround material spatial perception, the queen evolves

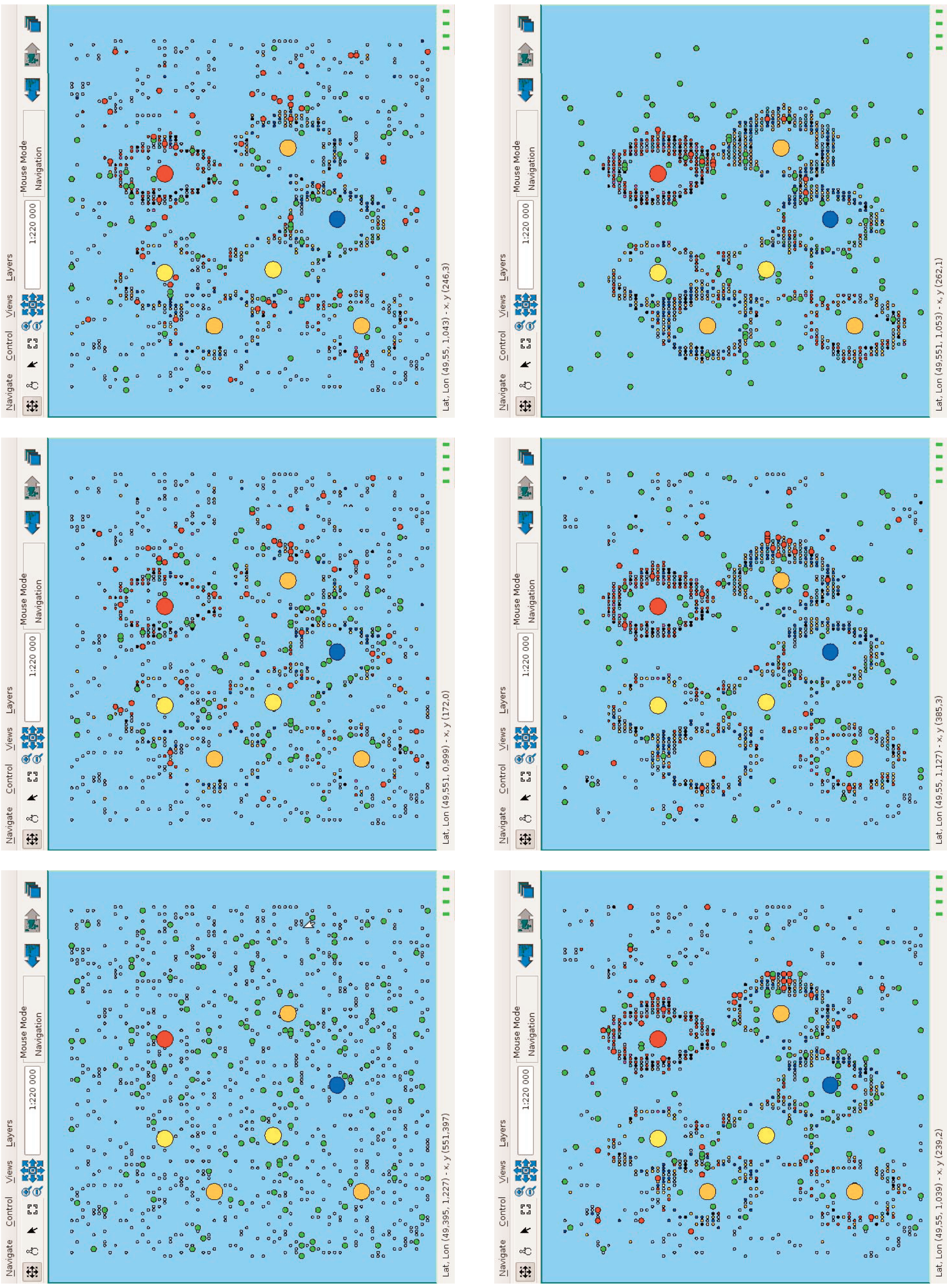


Figure 14: Simulation computation, at successive steps: iterations 0, 152, 250, 370, 601, 1601.

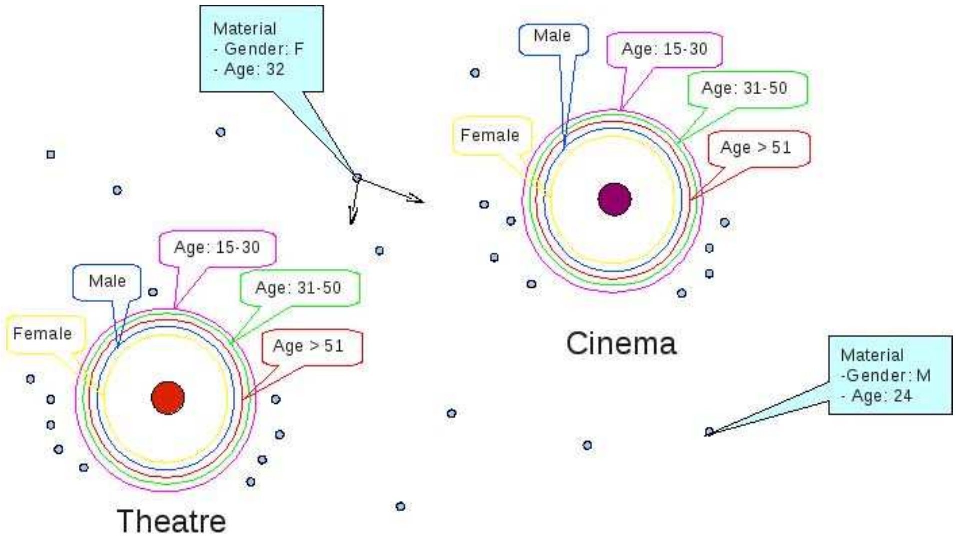


Figure 15: Cultural equipment dynamics modeling

traversal mode where they wait a given number of steps proportional to the length of the edge. The time to wait on the edge is computed according to the real road length and its speed limit.

Each ant owns a tabu list containing already traversed edges. When an edge is crossed it is put in the tabu list. The list is a FIFO stack, and therefore ants forget edges after a given number of steps. When an ant chooses the next edge it will cross, it only considers edges that are not in its tabu list. If there are no possible edge the ant is moved to a random node in the graph. This happens when either all edges are in the tabu list, or when the ant reached a dead-end, or the frontier of the area described by the graph. This allows to model the fact a vehicle left the graph (by a leaving road or because it stopped), and to make another vehicle start a journey in the graph at a random point.

Each time an ant finished crossing an edge, it drops a small quantity of pheromone on the edge. These pheromons evaporate at a constant rate. Therefore, after a period of initialization, and because ants move at random, the level of pheromone on edges will stabilize and the most used roads will have more pheromons that the less used ones.

Once again this algorithm use only local decisions at the level of the ant, and no global control is necessary. Therefore it can easily be distributed for large simulations on dense road network areas.

The figure 16 show the simulation on the city of Le Havre. This city is a harbor with the sea at the south and west. Therefore all roads leading to the city come from the east and north. The pheromon quantity on roads is shown by color. Blue means very few pheromons, yellow to red means a lot of pheromons. The scaling from blue to yellow to red is logarithmic with full red the maximum level during the simulation. The two main access points to the city are the two highways A29 and A13 that are both highlighted in red as well as the main city entry the Winston Churchill Avenue. The city centers are highlighted in yellow. The Breque cross road which is one of the most used way when coming or leaving the city is also highlighted in red.

Figure 16: Random Walk algorithm applied on Le Havre city road network

5. Conclusion

In this tutorial, self-organization mechanisms are described in natural systems and are used to design algorithms for distributed computing. Applications on ecosytems or urban dynamics modeling are presented. A processus to distribute decentralized simulations from its communication graph is also presented. Standard algorithms extensions need to achieve these processes or simulations are developed. The decentralized approaches, inherent to these bio-inspired algorithms, allow to extend them, respecting the complexity of the phenomena to model.

References

[1] M.A. Aziz Alaoui and Cyrille Bertelle (eds) (2009) *From System Complexity to Emergent Properties* “Understanding Complex Systems” series, Springer-Verlag, Berlin-Heidelberg.

[2] M.A. Aziz Alaoui and Cyrille Bertelle (eds) (2006) *Emergent Properties in Natural and Artificial Dynamical Systems* “Understanding Complex Systems” series, Springer-Verlag, Berlin-Heidelberg.

[3] M. Batty and Y. Xie. From cells to cities. *Environment and Planning B*, 21:531–548, 1994.

[4] Michael Batty. *Cities and Complexity*. The MIT Press, 2005.

[5] Eshel Ben-Jacob and Herbert Levine. The artistry of nature. *Nature*, 409:985–986, 2001.

[6] Itzhak Benenson. Modeling population dynamics in the city: from a regional to a multi-agent approach. *Discrete Dynamics in Nature and Society*, 3:149–170, 1999.

[7] Itzhak Benenson and Paul M. Torrens. *Geosimulation*. Wiley, 2004.

[8] Geraldo Beni and Jing Wang. Swarm intelligence. In *Proceedings Seventh Annual Meeting of the Robotics Society of Japan*, pages 425–428, 1989.

[9] Bertelle,C.; Duchamp, G.H.E. and H. Kadri-Dahmani (eds) (2009) *Complex Systems and Self-Organization Modelling*, “Understanding Complex Systems” series, Springer-Verlag, Berlin-Heidelberg.

[10] Bonabeau, E.; Dorigo, M. and G. Theraulaz (1999) *Swam Intelligence, from natural to artificial systems*, “Santa Fe Institute Studies in the Sciences of Complexity” series, Oxford University Press.

[11] E. Bonabeau, G. Theraulaz, J. Deneubourg, N. R. Franks, O. Rafelsberger, J. Joly, and S. Blanco. A model for the emergence of pillars, walls and royal chamber in termite nests. *Phil. Trans. R. Soc. Lond. B*, 353:1561–1576, 1998.

[12] Eric Bonabeau, Guy Theraulaz, Jean-Louis Deneubourg, Serge Aron, and Scott Camazine. Self-organization in social insects. *Trends in Ecology & Evolution*, 12(5):188-193, 1997.

[13] Anne Bretagnolle, Eric Daude, and Denise Pumain. From theory to modelling: urban systems as complex systems. *Cybergeo*, 335:26 pages, 2006.

[14] Camazine, S.; Deneubourg, J.-L.; Franks, N.R.; Sneyd, J.; Theraulaz, G. and E. Bonabeau (2001) *Self-Organization in Biological Systems*, Princeton University Press.

[15] Andrew T. Crooks. The repast simulation/modelling system for geospatial simulation. Technical report, UCL Working Papers Series, September 2007. Paper 123.

[16] Antoine Dutot. *Distribution dynamique adaptative à l'aide de mécanismes d'intelligence collective.* Phd thesis, University of Le Havre, 2005.

[17] Rawan Ghnemat (2009) *Adaptive Modeling for Spatial Emergence within Complex Systems*, PhD Thesis, Le Havre University, France.

[18] Ghnemat, R.; Bertelle, C. and G.H.E Duchamp (2007) *Adaptive Automata Community Detection and Clustering, a generic methodology*, in *Proceedings of World Congress on Engineering 2007, International Conference of Computational Intelligence and Intelligent Systems*, pp 25-30, London, U.K., 2-4th July 2007.

[19] John Holland. *Adaptation in Natural and Artificial Systems.* Ann Arbor, 1975.

[20] James Kennedy and Russ Eberhart. Particle swarm optimization. In *Proceedings of the 1995 IEEE International Conference on Neural Networks*, pages 1942–1948, 1995.

[21] James Kennedy and Russel C. Eberhart. *swarm Intelligence.* Morgan Kaufmann Publishers, 2001.

[22] Charles M. Macal and Michael J. North. Tutorial on agent-based modeling and simulation. In *Simulation Conference*, 2005.

[23] Zachary Mason. Programming with stigmergy: Using swarms for construction. *Artifcial Life*, VIII:371–374, 2002.

[24] Robert Najlis and Michael J. North. Repast vector gis integration. In *NAACSOS Conference (North American Association for Computational Social and Organizational Science)*, Notre Dame, Indiana, USA, June 26-28 2005.

[25] Denise Pumain, Fabien Paulus, Céline Vacchiani-Marcuzzo, and José Lobo. An evolutionary theory for interpreting urban scaling laws. *Cybergeo: European Journal of Geography (on-line)*, page article 343, 2007.

[26] Vitorino Ramos, Fernando Muge, and Pedro Pina. Self-organized data and image retrieval as a consequence of inter-dynamic synergistic relationships in artificial ant colonies. In *Soft Computing Systems - Design, Management and Applications, 2nd Int. Conf. on Hybrid Intelligent Systems*, volume 87, pages 500–509. IOS Press, 2002.

[27] Martin Raubal and Claus Rinner. Multi-criteria decision analysis for location based services. In *Geoinformatics 2004, Proceedings 12th Int. Conf. on Geoinformatics - Geospatial Information Research: Bridging the Pacific and Atlantic*, University of Gavle, Sweden, June 7-9 2004.

[28] Repast, See in march 2008. http://repast.sourceforge.net/.

[29] Craig W. Reynolds. Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics (SIGGRAPH '87 Conference Proceedings)*, 21(4):25–34, 1987.

[30] Ajay Sharma, Vishnu Vyas, and Dipti Deodhare. An algorithm for site selection in gis based on swarm intelligence. In *IEEE Congress on Evolutionary Computation, CEC 2006*, pages 1020 – 1027, 2006.

Professor Cyrille BERTELLE
LITIS - University of Le Havre
25 rue Philippe Lebon, BP 540
76058 Le Havre Cedex, FRANCE
email:*cyrille.bertelle@gmail.com*

Doctor Antoine DUTOT
LITIS - University of Le Havre
25 rue Philippe Lebon, BP 540
76058 Le Havre Cedex, FRANCE
email:*antoine.dutot@gmail.com*

Doctor Rawan M. GHNEMAT
German-Jordanian University
PO Box 35247
Amman 11180, JORDAN
email:*rawan.ghnemat@gju.edu.jo*