22

Collective displacement of modular robots using self-reconfiguration

Carrillo Elian[1,2] and Dominique Duhaut[1]
[1]University of Bretagne Sud, University Romulo Gallegos[2]
France & Venezuela

1. Introduction

Collective displacement is a very useful behaviour for living creatures. This behaviour can appear in a flock of birds, a school of fish, or a swarm of insects. Flocking behaviour is a common demonstration of the power of simple rules in collective displacement emergence by (Reynolds, 2007).
The study of the displacement of a robot in an unknown universe is a traditional subject of robotics (Fredslund & Mataric, 2002). We address the problem of the displacement of a group of robot modules which are part of a reconfigurable robot **(Christensen, 2005)**. Collective displacement is considered a very complex problem (Yoshida, 2001). The number of possible solutions gives a combinative explosion in the graph of possible displacements.
In this chapter we target the collective displacement through modular self-reconfigurablility. The objective is to find simple rules to co-ordinate autonomously a high level of decision of action for the modular robot. By the implementation of those same rules in all the modules of the robot an emergent displacement should appear. This emergent displacement will be used to reach the goal.

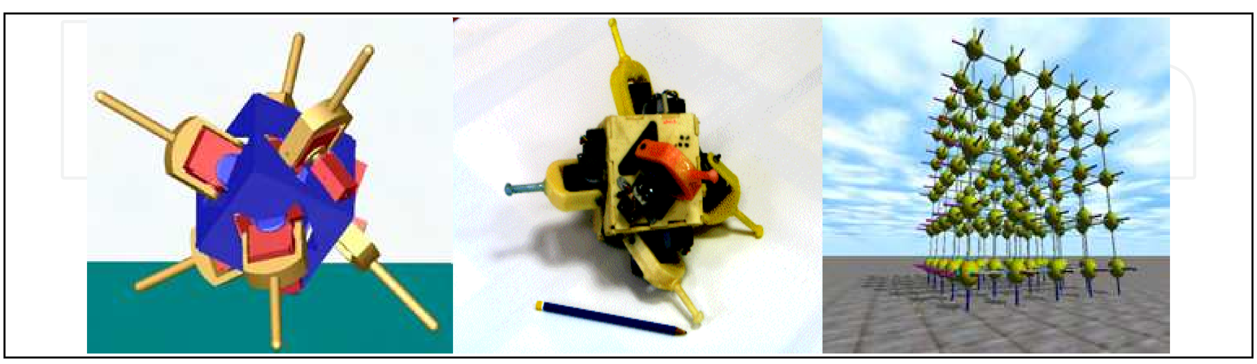


Figure 1. MAAM, modular and reconfigurable robotic project

Here we consider the modular self-reconfigurable robots developed in the MAAM project. It's a homogeneous and self-reconfigurable multi-robot system where all the robots have the same competencies, same perceptions and same capacities. Thus the model of making the self-reconfiguration is based on some capacities of the modular MAAM robots.

Source: Bioinspiration and Robotics: Walking and Climbing Robots, Book edited by: Maki K. Habib
ISBN 978-3-902613-15-8, pp. 544, I-Tech, Vienna, Austria, EU, September 2007

2. Problems in collective displacement

Here we address the problem of collective displacement of a reconfigurable robot and we seek to define the unique program to be loaded up in each module of the robot (Duhaut & Carrillo, 2007).
The main difficulties of this problem are to not separate elements of the robot during reconfiguration, and to find a method ensuring a global solution in a reasonable time without explicit communication.

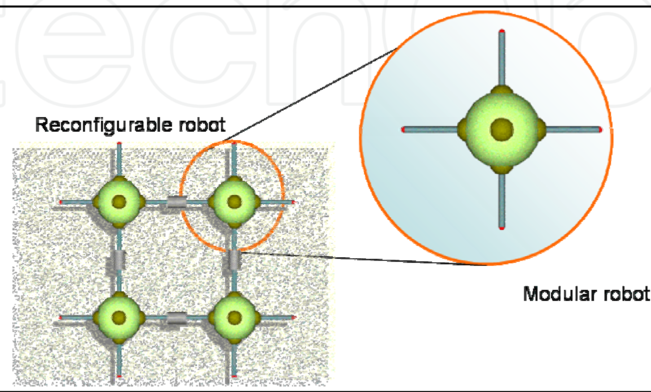


Figure 2. Reconfigurable robot and nodular robot

Connection is a constraint related to the nature of the robots used. A reconfigurable robot consists of a whole of modules connected between them. In our case, each module is a robot provided with actuators, sensors and a capacity of decision. We seek to maintain the connection of the modular robots during reconfiguration, so that the reconfigurable robot "never breaks".
A rapid calculation of complexity shows the level of difficulties to resolve this problem in a reasonable time. Let us consider that if we have N modular robots and each modular robot can make 8 different movements then at a given moment we could sight 8^N possibilities of movements. So now if we consider that each movement approaches the goal of a unit and that the goal is at a distance P, then the total number of possible movements is of $(8^N)*P$, e.g. let us say that the reconfigurable robot is composed of ten or twenty modular robots and a hundred steps of distance from the goal. The calculation becomes:

$$(8^{10})*100 = 1x10^{11}$$
$$(8^{20})*100 = 1.1x10^{20} \quad (1)$$

This number of possibilities is out of the range for motion planning in current computers. In fact, different studies have been made around programming reconfigurable robots in the MAAM project. The limitation to resolve this problem using mathematical resolution systems is that the number of degrees of freedoms in the reconfigurable robot is too large and has infinity possible resolutions. Using other techniques like Markov Decision Process or a Learning Classifier System (MAAM, 2007) the space needed for the storage information is not enough to make a global description of the problem.
The major problem is that the combinatory explosion of the possibilities gives plenty of local minimal solutions. The exponential order of the number of possibilities shows that resolve this problem using a centralized planning decision control command could be very difficult to find.

We propose an emergent solution to resolve the problem of collective self-reconfiguration. Using a distributed approach based on the co-operation of the modular robots using a reactive decision order. The solution is obtained by the emergence of all the modular behaviours, and visualized the reconfigurable robot behaviours as the addition of local robots-decision level (Carrillo & Duhaut, 2005). Limiting the decision-making process to a purely local situation also makes possible to reduce the complexity of the program.
For this reason we seek to define a single program for each modular robot. This program will be the individual behaviour for each one of them. This behaviour will make a local decision according to the capacities of perception of the immediate environment. The solution will become highly parallel since all the modular robots will make their calculations at the same time.

3. The simulated world of the robot

In this model, we suppose that all the modular robots have very small perception capabilities and detect only local environment. However, it can detect the direction of an attractor in its local neighbourhood. It can make the difference between a free space and occupied one. The model is assumed to be without explicit communication, in other words the robots cannot exchange messages with each other.
The reconfigurable robot is modelled by a multi-agent system. The environment of the robot is discretized in a 2D square grid model, represented in the form of a vectorial environmental. In this model, each cell is associated to its 8 neighbour cells. A robot replaces a cell in the matrix of the environment.
The destination to be reached for the robot is given by an attractor element which makes its possible to define a field of potential. The attractor is an element which emits a signal that can be recognized by the agents. In a first approach, the signal is distributed in the environment with an intensity which varies according to the distance. The further away the place in the environment from the attractor the weaker the signal will be. This signal is the potential field of the attractor which produces the internal gradient reward.

3.1 Regular potential field

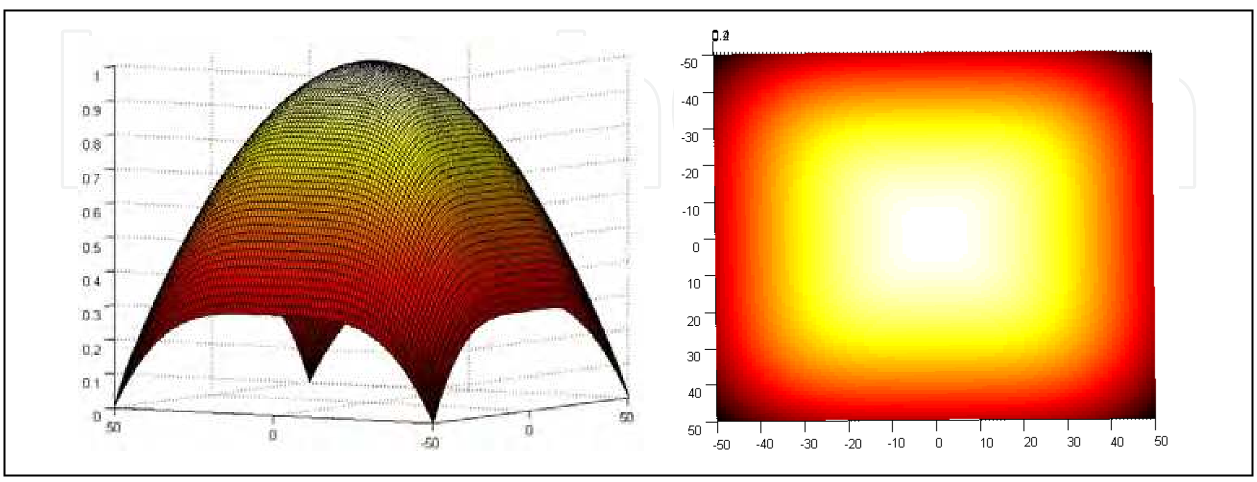


Figure 3. Regular potential field

The property of the regular potential is that all the points located at the same distance from the target measure the same value of the potential. This corresponds, for example, to a light bulb.
The agents build a representation of the environment and their surrounding 8 neighbours. They see their environments under various types of entries associated with the 8 possible directions of displacement in the plan.

3.2 The perception of the agents

By assumption the organs of perceptions of modular robots allow the agent: (a) to make a representation on the presence or the absence of another robot, (b) to measure the gradient resulting from the potential of the attractor. Both perceptions are made in the closer environment representation, that of the 8 neighbours.

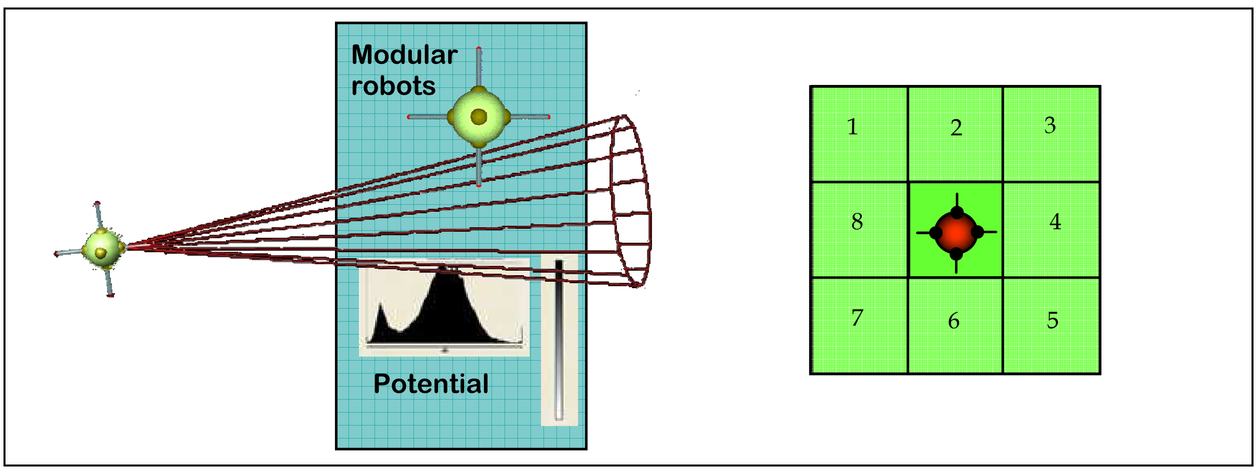


Figure 4. Robot perceptions and its 8 neighbours

3.3 Displacement of the modular robots

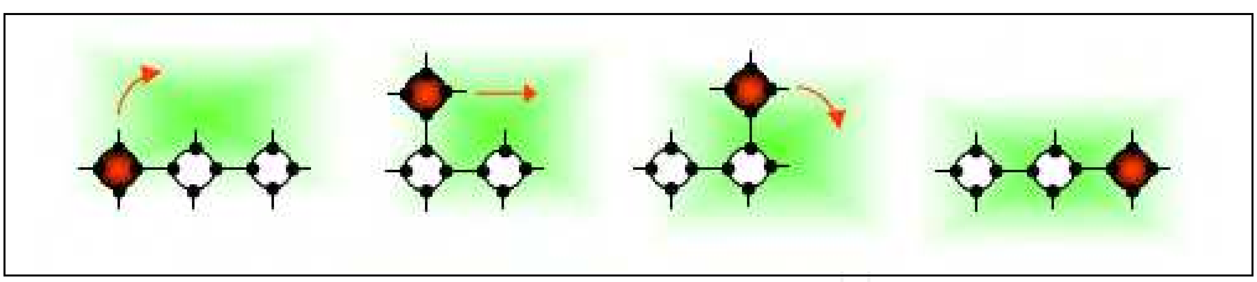


Figure 5. Example of reconfiguration

The agents are constrained to move towards a direction where an empty site is available (see figure above). The displacement of the agents is always made by the displacement of a connection from a robotic module to another. Connections, between the agents, are made through its 4 principal connectivities: North, South, East, West or directions 2,4,6,8 in Fig. 6. In the displacement of a connection to another, the robots do not have the possibility of moving other robots.

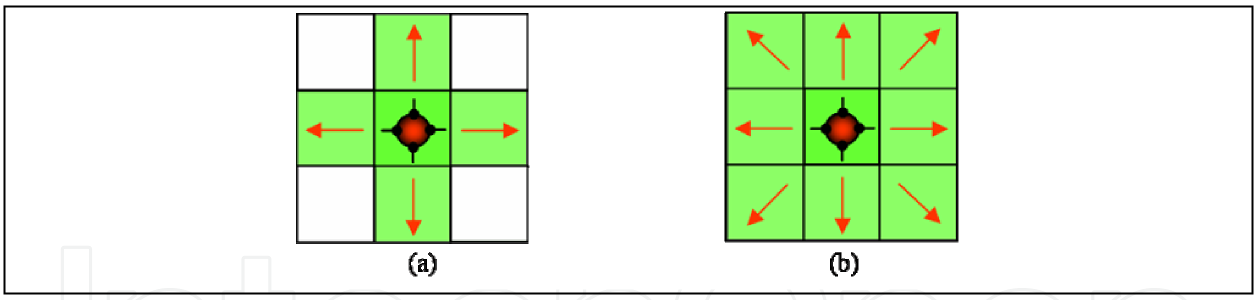


Figure 6. Robot possible (a) 4 connections, (b) 8 possible actions in neighbours

Here we find the ideas developed for the representation of the environment of a multi-agent system better known under the name of connectivity of Von Neuman and neighbours of Moore [Amblard 2006].
Hopefully, the restriction used so that the reconfigurable robot "never breaks" creates fewer options for the number of possibilities of actions. We have made an inventory of the possible configuration of other modules in the neighbourhoods. In this discretized world each modular robot appears as 0 or 1 in the 8-neighbour connection, so which gives $2^8 = 256$ possible configurations. Saying in other words, the authorized actions for the displacement of the modules are those which do not go against the cohesion of the group.
The constraints of displacements are: (a) to keep at least a connection with another agent, (b) if an agent keeps only one connection with the structure, it becomes a key agent for the cohesion of the group. It is not allowed to separate. To move, an agent does not have to acknowledge itself as a key part for the structure, (c) a place for a possible displacement must be empty, (d) the way for moving towards the position of destination must be free.

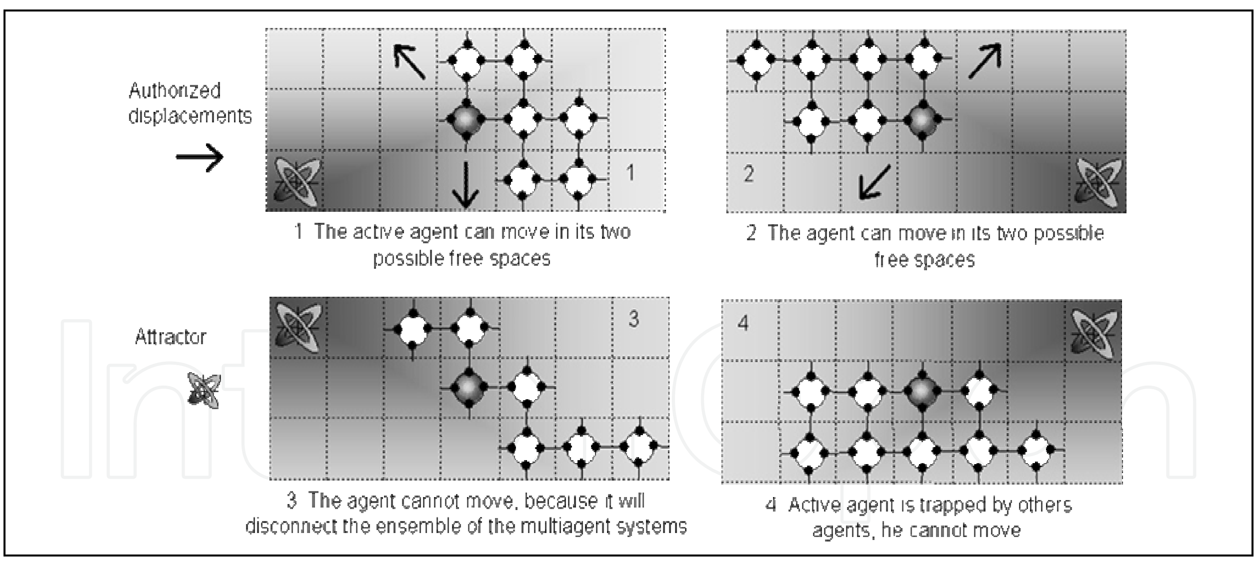


Figure 7. Example of authorized and unauthorized displacements

When an agent does not have more than 2 connections with another module, case 1 & 2 in Fig. 7., thanks to its vicinity, it knows that even if it moves it will not break the reconfigurable robot structure. Thus the movements are authorized. On the other hand, in case 3 the displacement of the robot would divide the reconfigurable robot in two, thus the movement is prohibited. In case 4 the movement is forbidden because we found that this type of movement starts creating holes in the structure, and generating a lot of problems in the global displacement.

3.4 The scheduling of calculations in the robots

The scheduling of calculations corresponds to the order in which the programs are performed in the modules of the reconfigurable robot. Of course, the ideal model is absolute parallelism: all calculations are done at the same time in each module. However, a reconfigurable robot is sequential by nature. Displacement cannot be carried out in parallel when two robots are close, because of the risks involved. It is supposed that the order of execution of movements of the robots is regulated at the mechanical level. A robot will move if it is guaranteed that the other modules do not move.
This constraint shows that on the level of the program, calculation is not the same if one makes the movement of a modular robot after or before another. To account for this problem we will suppose in the simulations that there are two types of possible software scheduling.
(a) Sequential Scheduling: In this case the modular robots receive a quota of time (presumably sufficient to make a complete iteration of the behaviour) one robot after another according to a fixed list. (b) Random Scheduling: In this case the modular robots receive their time quotas in a random way. This makes it possible to take into account the possible differences in the processing times of calculations in modular robots. Due to it being the most "chaotic" of scheduling, it forces us to develop sure methods of construction of the emergent algorithms.

4. Reactive programming

The objective of reactive programming is to reach a global displacement of the robot by the set of reconfiguration of all the modules. The constraint is that the robot must not lose any module during the global movement. We seek to define the program which must be charged in each robot module that makes robots move in an autonomous way towards a goal maintaining the cohesion of the group of modules.
This program will be the individual behaviour of each robot. This reactive behaviour will make a local decision according to the perception of the immediate environment. The interactions of all its modules in an autonomous way make the emergent displacement.

4.1 Forward algorithms

The principle of the reactive algorithm is to decrease the distance which separates the robot from the source of the attractor following a potential field (Arkin, 1998). Modular robots must take the decision of the place to go making a minimization of the distance with the information of the potential.
The algorithm decides the action to take according to two factors: the gradient of attractor and configuration of the modular robot in its neighbourhoods. Displacement is carried out towards a direction where the potential is augmented, respecting the cohesion of the reconfigurable robot structure.
In the simulations of section 5, we consider two different algorithms: (a) "total forward": In this case the module will make a continuation of displacement along the molecule until reaching a position which will be a minimum for the potential, (b) "one step forward": In this case the module will make only one displacement along the molecule then will stop and await its next quota of time. During this time all the other modules can move.

4.2 Blocking patterns

The major problems of a distributed emergent approach are the blocking and oscillation patterns. In this study we will show that these problems can appear at the time of the implementation of reactive behaviours. The problems which are necessary to confront are patterns in the structure that stop the progression of the reconfigurable robot to the goal.
Here we consider the problems of deadlock and oscillation appearing in collective displacement using self-reconfiguration without communication and only a local world representation. In section 5 we describe the reason of the emergent blocking patterns and next we bring a solution for this kind of collective displacement.

4.3 Sensitivity of the algorithms to the gradient

The descent of a simple gradient corresponds to the research of the direction of the displacement for which the gradient towards the target will decrease.
For the forward algorithms, the problem of oscillating modules in the structure can be solved adjusting the tolerance action decision (min-max) of the potential. This parameter setting is easy to make in simulated worlds, but the implementation in real robots takes more time and must be made more carefully.

5. Simulations of modular self-reconfigurable robots in a regular potential

Here we show some simulations of the reactive algorithms working as simple reflex agents. Different global actions emerge from the reconfigurable robot of 25 modules related to the physical initialisation or the order of calculation.

5.1 Total forward and sequential scheduling

In this first simulation we study collective displacement using a regular potential like a lamp which creates a regular gradient cantered at the point of the attractor.
Modular robots advance in a regular way each one in their turn with a sequential scheduling and the forward total algorithm. In this case, each module advances in the structure of the reconfigurable robot until it reaches a position with minimal distance to attractor.
The descent of the gradient is carried out continuously because sequential scheduling makes it possible and each module advance of all its possible movements with the forward total algorithm.
The emergent behaviour of the reconfigurable robot is to create a line in direction of the attractor. Each modular robot moves from a place in the structure to the place of better potential, at the end of the line in direction of the attractor.

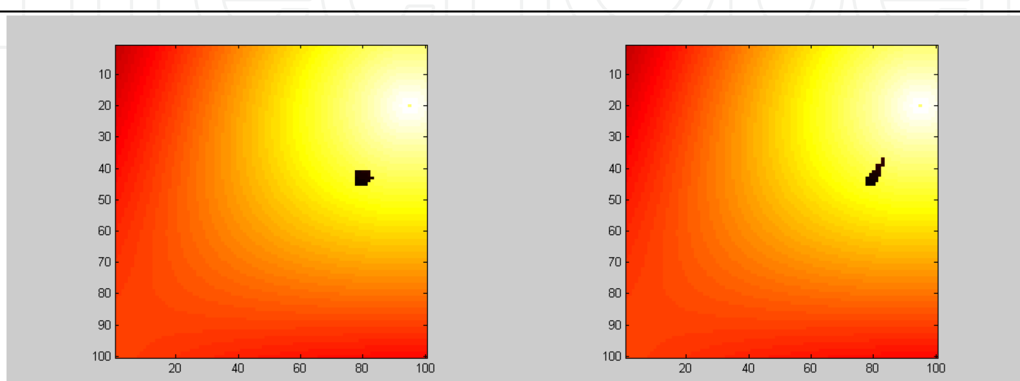


Figure 8. (a) Total forward with a sequential scheduling in a regular potential field

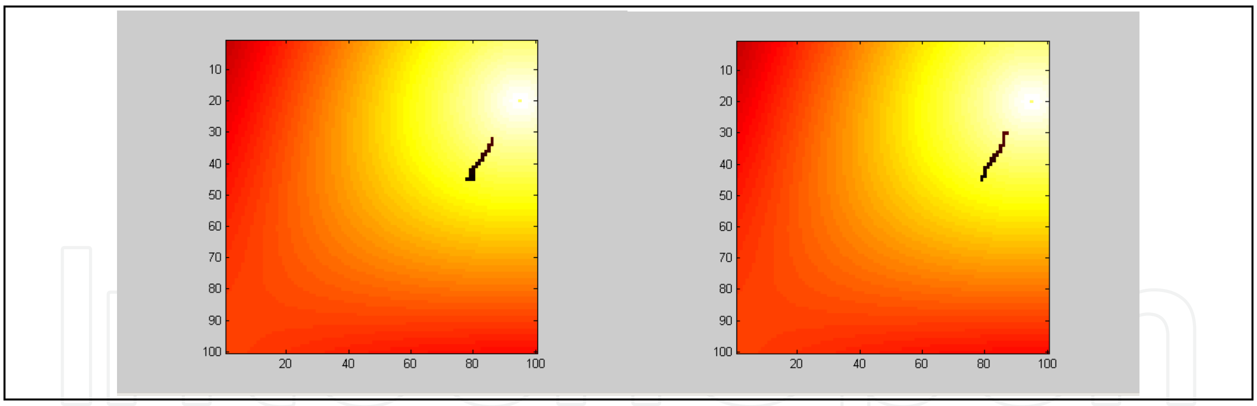


Figure 8. (b) Total forward with a sequential scheduling in a regular potential field

In this type of simulation the results are always satisfactory when a line is formed and the modular robots roll around each other in direction of the target.

5.2 Total forward and random scheduling

But as we can appreciate in Fig.9, changing initial conditions like the position of the reconfigurable robot, or the number of modules, the global behaviour can changes.

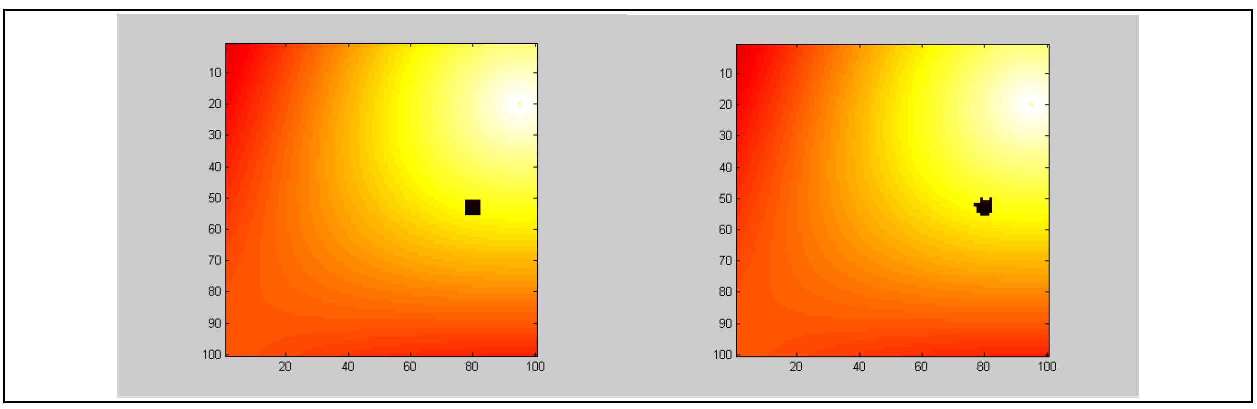


Figure 9. (a) Emergence of U with forward total and sequential scheduling in a regular potential field

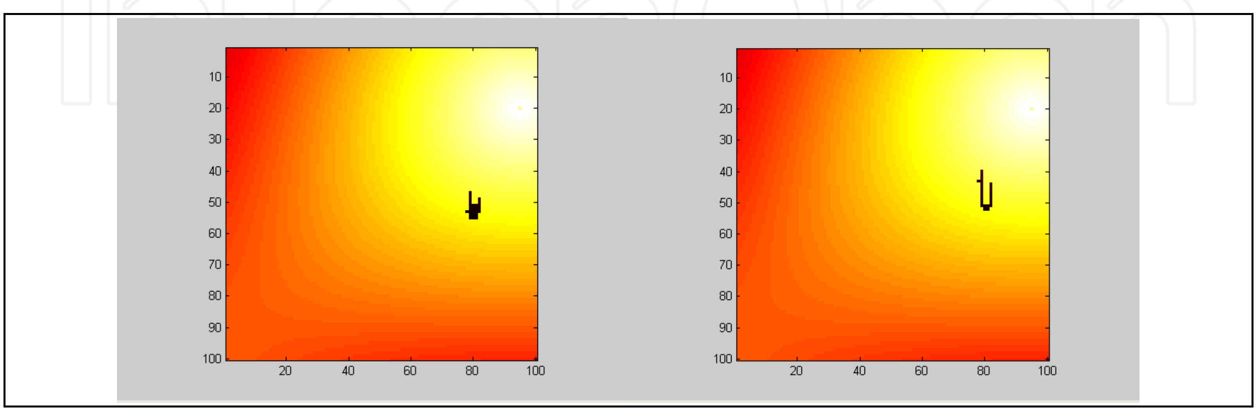


Figure 9. (b) Emergence of U with forward total and sequential scheduling in a regular potential field

With random scheduling, results are not better than in sequential scheduling, in fact the results are almost the same. The forward total ensures that all the atoms reach their best place in the structure.

5.3 One step forward and random scheduling

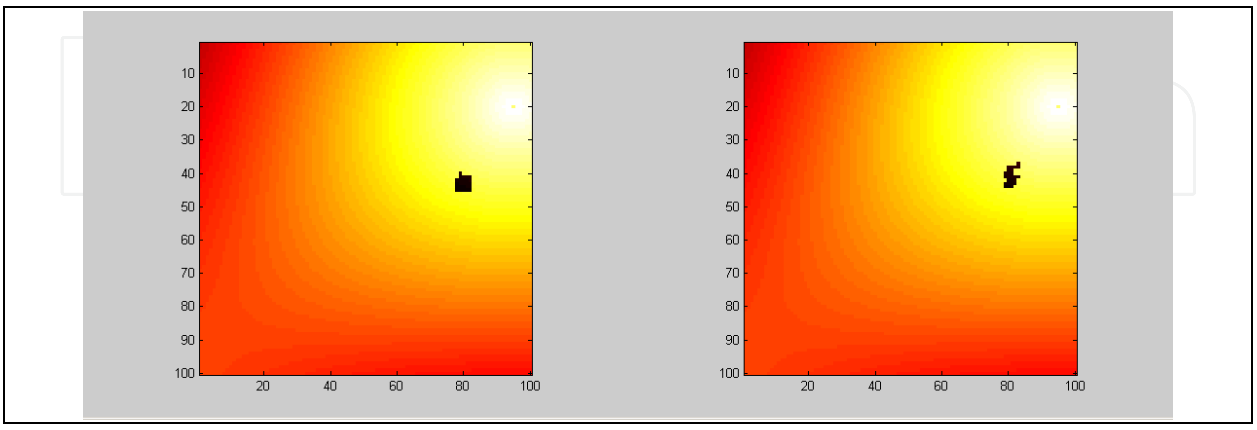


Figure 10. (a) Emergence of U and O with one step forward and random scheduling in a regular potential field

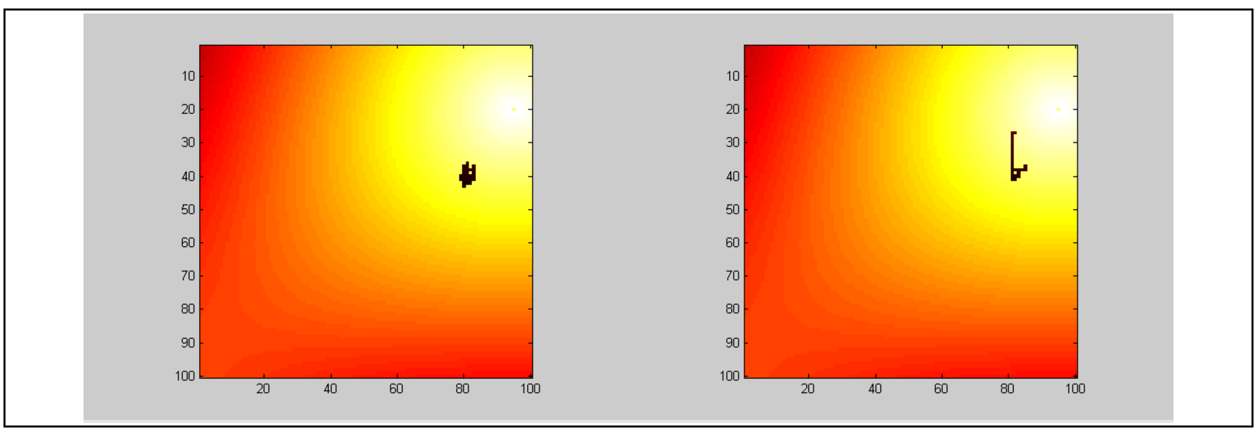


Figure 10. (b) Emergence of U and O with one step forward and random scheduling in a regular potential field

Similar results are given by using sequential scheduling**.**
The physical constraints are at the base of some problems which emerge in algorithms used for most multi-agent displacements (Arkin, 1998): deadlock and oscillation. In fact methods for collective displacement are not adapted for collective displacement by self reconfiguration. We will show that these two principal problems can appear in the implementation of such an approach. We will show in particular that simulations which "go to goal" hide heavy defects for displacement by self-reconfigurable. We will be able to prove that this type of approach is sensitive to the scheduling of the decision-making in the modules.
The generation of deadlock created by the blocking patterns in the structure of the reconfigurable robot stop the progression. Physical factors and scheduling order are involved in this problem for the displacement of the group by self reconfiguration.

The emergent behaviours are different with the algorithm "total forward" or with the "one step forward", especially in the blocking parameter causalities. In fact, the further the position to reach to the reconfigurable robot, the more the "one step forward" has the chance to arrive to a minimal local configuration (stopping the collective displacement) in the numbers of possible combination of modules actions.

6. How to avoid blocking patterns

After analyzing the simulations and the obtained emergent results, we can say that there are two principal problems to be solved during collective displacement. One, due to the nature of the structure and the representation of the potential field, and the other, related to the dynamics of execution.
To find a solution for our problem of collective displacement we must solve those two problems. From the nature of the structure, we can remark that the absence of a unique best place of the potential resulting from the attractor could generate a blocking pattern. On the other hand, it has been shown that the dynamics of execution could generate blocking patterns if a module could not reach its position of minimum of algorithm minimization in the progression being blocked by another module.
As we can see in preceding simulations, it is really easy to find blocking patterns in the U or O structures by using random scheduling in a regular potential field. Avoiding this problem is convenient to use a potential with other proprieties.

6.2 Double potential field

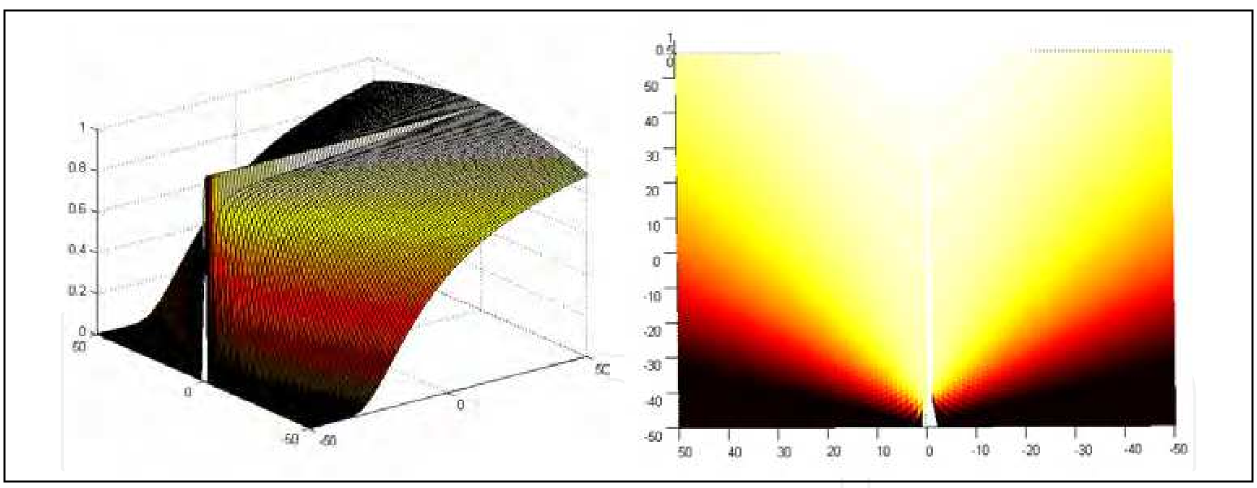


Figure 11. Double potential field

The property of the double potential is that it decreases in two dimensions, with a line of stronger intensity which corresponds to a spot of light.
There is a line D passing through the attractor where the intensity is maximal on the potential field. The signal decreases regularly in two directions. The line of stronger intensity decreases according the distance to the attractor. The perpendicular potential from the line of stronger intensity decreases in a Gaussian form.
The equation of Gaussian with an axis of symmetry can be written in the following form:

$$f(x;\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{2}$$

In the Fig. 12, σ evolves according to the X-coordinates: the more the X moves away from attractor and greater σ becomes. With this parameter and keeping C=0 constant, we model a potential filed shape of a hull of boat or a cone of light and with a line of stronger intensity.
With such a type of potential, the knowledge of the potential of an unspecified point M and its values of the potential in the neighbourhood make it possible to define the position in space. Thus we can know from a point M to the line of stronger intensity and the distance to the attractor.
The representation of this potential model in captors is to have two measuring parameters who defined a distance that ensure a unique point P in the structure.

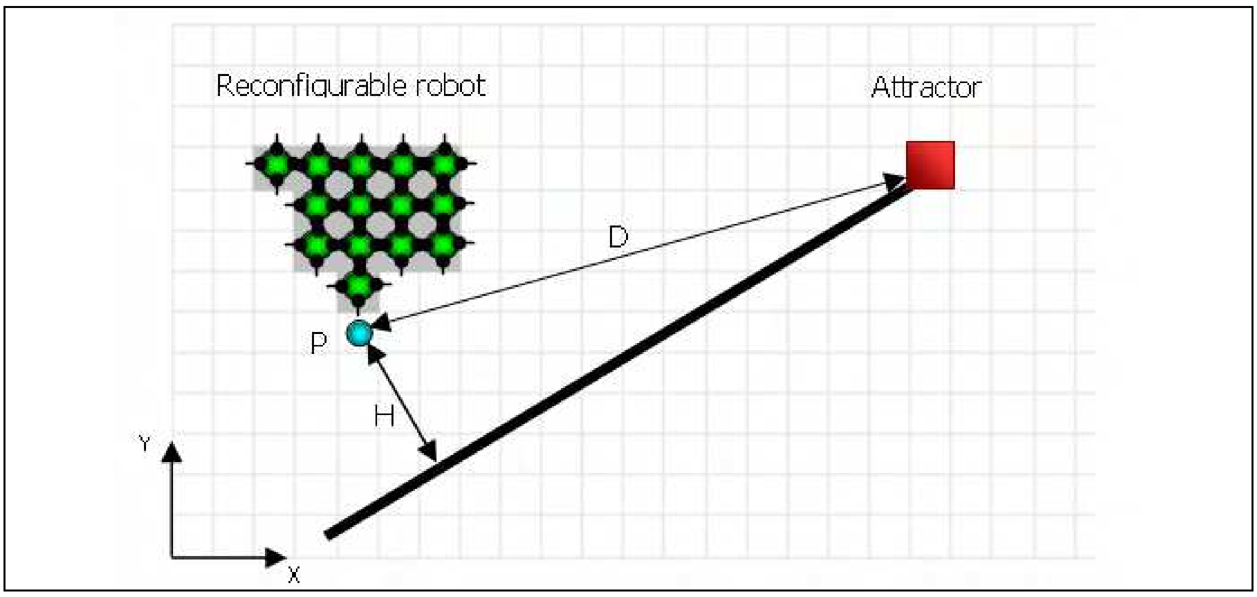


Figure 12. Distance D and H of the P point

For the descent of the double gradient it is necessary to make the difference between two kinds of properties: a) there are directions of possible movements for module by following axis X and Y, b) there are two sizes, D which measures the distance from the transmitter and H which measures the distance from the line of stronger intensity.
The descent of the double gradient, based on an order total given by a calculation of the distance on the couple (D,H). In this case a movement in X or Y will be considered as acceptable if the new position (D', H') checks the condition:

$$(H,D) > (H', D') \text{ if } (H > H') \text{ or } (D > D' \text{ and } H = H') \tag{3}$$

We can think to permute the order of evaluation of the distance (D,H) by (H,D). But in fact blocking patterns could reaper in the system, due to the fact that D is like the radial rang value of a regular potential.
The double potential is a solution for this problem. Indeed, these kinds of potential have two variables for representation in 2D space. All the points in the 2D space are unique by their correspondence of the attractor and the line of stronger intensity
In such a potential, there is a line of stronger intensity where potential changes uniformly according to the distance with the attractor transmitter. Each point is single by its vicinity.

For the reactive model by distance decision we take the following measuring parameter: a) the distance from a point P to the line of stronger intensity is H, b) the distance with the target is D.

$$(H,D) > (H', D') \text{ if } (H > H') \text{ or } (D > D' \text{ and } H \approx H') \quad (4)$$

In equation 4 the relation “>” is a relation of order. The “≈” is due to the introduction of tolerance parameter to follow the line of stronger intensity.

6.3 A methodology to guarantee convergence

To avoid the appearance of fingers during the execution of the displacement of the reconfigurable robot we have to prevent that for some configurations, the reconfigurable robot has several minimal positions created by the succession of elementary module displacement. Supposing that the initial conditions of the reconfigurable robot check the properties described below.

Property 1: a unique minimal point

For related configurations of the reconfigurable robot without holes in the reconfigurable robot thus without "O" and initially compacts (in the shape of a square, rectangle, rhombus...), for a modular robot with the position M taken on the periphery of the reconfigurable robot (thus able to move) it exists for the modular robot a P minimal and unique to reach, starting from M to P.

In the example in Fig. 12, the point P is the point to reach for all the modular robots able to move in the reconfigurable robot.

This property makes it possible, to not have several local goals to reach in the initial configuration of the reconfigurable robot. This is obtained by using the equation 4 for the descent of the double gradient.

Property 2: active modular robots

The elementary movements programmed in the reactive algorithm authorize the displacement of modules that are at the end of a line and a column. These modular robots are called active modular robots.

The authorized movements of M to M′ check M′ < M.

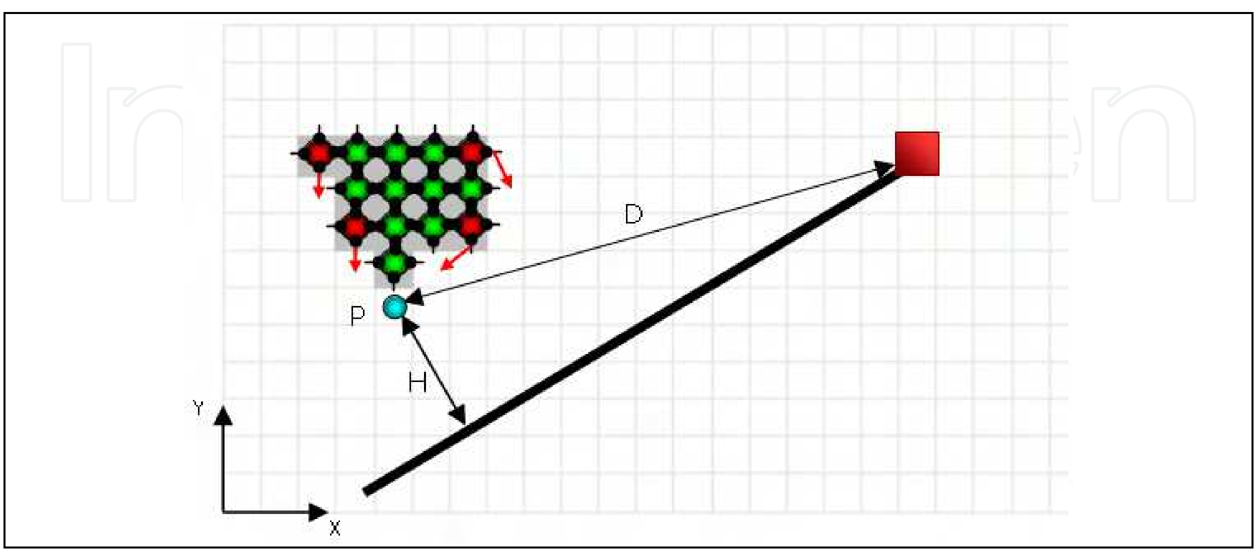


Figure 13. Modular robots with reactive algorithm movements

In the example, its show that only four modular robots check the conditions above, and the arrows indicate the authorized movement.
This makes it possible to avoid the formation of "holes" in a line of modular robots. In fact certain carried-out simulations showed that there could be convergence by slackening this constraint.

Property 3: path decreasing at the minimal point
For all the modular robots checking property 2, there is a single path S decreasing from M to P. The path is defined as a finished succession of authorized movements.

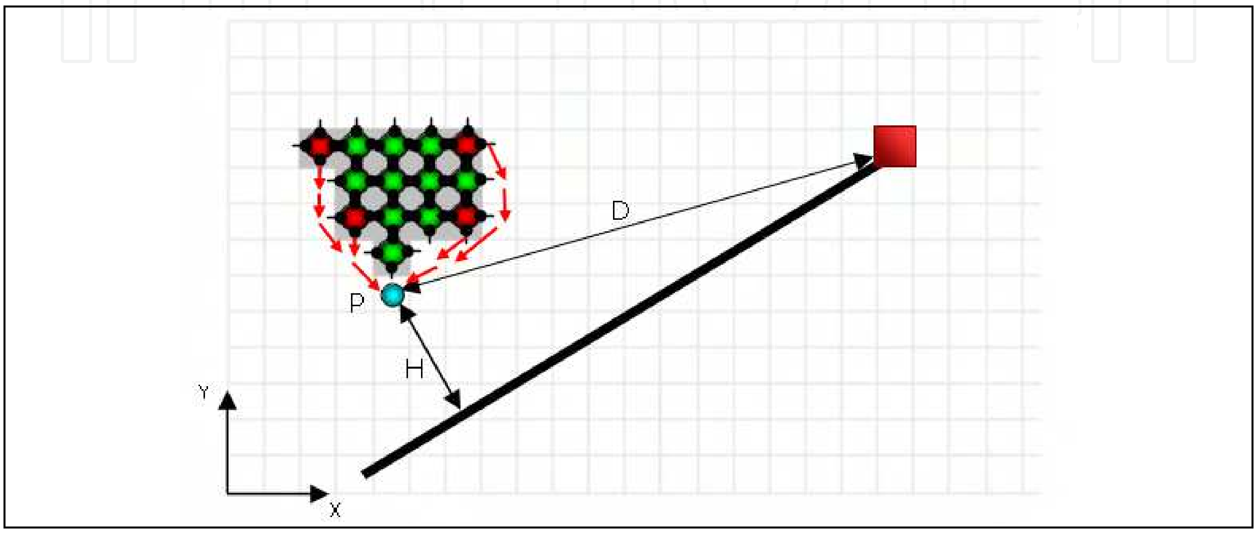


Figure 14. Decreasing paths

This property of regularity of the structure makes it possible to prevent the reconfigurable robot from containing in its configuration a "bump" which could generate the formation of a finger

Property 4: progression
A modular robot moves along its path from M to P only it that does not block another active modular robot.

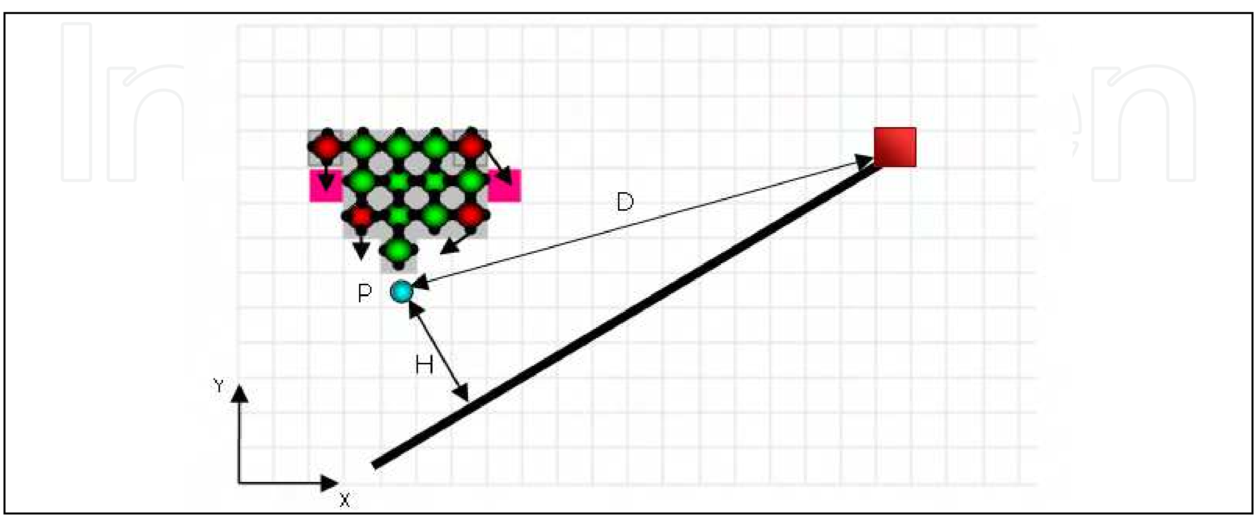


Figure 15. Progressions

This property must be checked at the initialization of the program, we will show that it remains maintained during the execution of the program. The programming of elementary displacements allows the active modular robots which are ahead (within the meaning of the order on the environment) to block the advance of those which are behind.
In the example above the two modular robots at the top in red will not be able to exceed the pink places, because, if they overtake the other robots in red they could block the progression of the reconfigurable robot by the creation of a blocking patterns.
This property forces the scheduling of the movements of the modular robots. It implies that an unspecified modular robot is able to determine if the modular robots with which it progresses have or not a path with the partial target P or not.

Lemma 1
If an active modular robot reaches P then the new minimal point P' is unique.
Proof:
As the point P is minimal then the P' point is close to P because of the regularity of the order. This point exists since the environment uniformly decreases (except on the target where the program stops).The P' point less than P is thus unique since P is unique and minimal by property 1 and that there are not two identical points in the environment.
QED

Lemma 2
The new P' point is accessible by all the active modular robots from the reconfigurable robot.
Proof:
By property 2 all active modular robots have the possibility of reaching P by a path uniformly decreasing. As P' is close to P, the access to the P' point thus obliges to replace the last movement towards P by a succession of two movements, the first to reach P, the second to reach P'. It is supposed here that the elementary movements programmed in the modular robots make it possible to solve all these configurations of displacement from P to P'.
The decrease of the path is obvious since P' is minimal by lemma 1.
QED

Theorem
If the four preceding properties are then checked the reconfigurable robot will converge towards its goal whatever the order in which calculations will be carried out.
Proof:
This proof consists in show that for each intermediate goal P:
1. there is an active modular robot M which will reach P.
2. all the other active modular robots will be able to reach the intermediate goal following P'.
3. the new active modular robots which will be created following the displacement of M will have properties 3 & 4 and will be able to reach P'

if such is the case then as there is a path C decreasing of P initial until the attractor by construction of the potential, there will always be a modular robot to reach the intermediate points of C without blocking since property 4 is checked at every moment.
1. there is an active modular robot M which will reach P:

The position P will be necessarily reached by an active modular robot. Because of the property 4 only those which have a free path to P will be able to reach it. If there are 2 modular robots in 2D space, one coming with P from the left and the other from the right. It is not possible to predict which will reach P due to it dependence on scheduling. On the other hand it is known that one of both will indeed reach the target since by property 4 nothing will be opposed to its displacement.

1. is checked.
2. all the other active modular robots will be able to reach P':

For all those which could reach P by lemma 2 can reach P'. By property 4, no one is blocked in its progression thus (2) is checked.

(3) the new active modular robots which will be formed by the displacement of M will have properties 3 & 4 and will be able to reach P':

As the modular robot M which will reach P is an active modular robot, then by definition, it is the end of a line and a column. When a module start the displacement to reach the target, necessary let free its old place. Since there is a discretized environment at less one neighbour robot take the place of the new end of a line and a column.

This makes that the active modular robots in M′ created by the displacement of another robot from M, can position in the place of M in a few displacements or its following path. Since they occupied the position of M or the following path, the modular robots in M′ check property 3.

Lemma 2 also makes it possible to know if these new active modular robots can also reach P'.

By the programming of the behaviours of the modular robots, we will suppose (as previously) that the new active modular robots check property 4.

(3) is checked.

QED

This constraining methodology makes it possible to guarantee the convergence of the reconfigurable robot whatever the number of modular robots or position of the target.

6.4 How to force a scheduling

As seen before, there are modular robots (at least one) which have a decreasing path through the reconfigurable robot. The phenomena resulting from the dynamics of execution show us that blocking patterns happen when modular robots cannot continue to the position P because they are blocked by another modular robot. Because some modules can have different calculation times in random scheduling, they can overtake other active modular robots in the structure. To prevent this kind of interaction and force the scheduling of the calculation to follow the decreasing paths to the point P, we propose a method that guarantees the emergence of the desired solution.

The solution suggested here is to add a state and a perception in the modules, a) to add a new physical state for the expression of an internal state which would be easily detectable by the other modules. The module must have a detectable characteristic which makes it possible to give it a Boolean value state. This Boolean value makes it possible to express in the vicinity which one is "active or not". The active modular robots of the reconfigurable robot will be in the high state “1” and the others will be in the low state “0”. b) to add a perception, in order to be able to detect the vicinity, we suppose that there is a sensor (or a method) making it possible to know the presence or the absence of a neighbour. This capacity of detection of presence is add as the possibility of determining the state of the neighbour by the reading of Boolean information. Once the state of the neighbours is

known, one will be able to progress towards the target if there is no modular robot in active state in the direction of the path.
Notice: we can note that the introduction of this information on the state of a module is related with communication. Indeed, a module can "say" to its entourage that it is "in this state". However, (as are the marks of pheromones) it acts like a form of marking of the environment and of an indirect communication. It is only detectable locally by another module.

6.5 Algorithm

In the continuation we use the simplicity of the algorithm "one step forward" with the new restrictions of movements to avoid the blocking of an active modular robot.
Algorithm "one step forward with perception of state":

if there are positions maximizing the potential
then to check for these positions the accessibility using perception of state;
to choose the best action; (5)
to move
if not to not move

In this case the module will make only one displacement along the reconfigurable robot then will stop and await its next quota of time. During this time all the other modules can have moved. If a modular robot is active its movements can be delayed by another active modular robot, but that does not mean that it will be the cause of a change of state to passive.

An active modular robot will not interfere with another active by requesting it for a reconfiguration. It will not crossover another active module, since that would block the continuation of its objective.

6.6 Simulations with perception of state

In this simulation the emergence of the desired behaviour of collective displacement towards the attractor appear under random scheduling. About sixty modular robots are simulates in a square environment of a hundred cells long.
The system of a global displacement towards the attractor emerges in the reconfigurable robot. It needs a time of transition so that all the modular robots change behaviour (when ($H>H'$) decision change to [($H \approx H'$) and $D>D'$] in distance minimization) once reached the line of stronger intensity at the right of Fig. 16.(b).

7. Discussion and future research

This study corresponds to futures steeps in the development of modular robots in MAAM. If technology continues to evolve as well it has done, we can imagine other project can be carried out again in micro-robotics and nanotechnologies. The challenge could be to reconfigure thousand of units under environmental conditions.
We wonder about the generalization of the approach and the portability of this method in particular if one moves to 3D space. Indeed, it is necessary to add a new Z referring distance parameters, and to define the order of minimizing the distance on X, Y, Z.

Another study would be related to the tolerance of fault. Multi-agent systems have the property of being resistant to the disturbances because of the great number of agents which cooperate. The question would be to know how to integrate the faults resistance of the reconfigurable robot without losing the property of convergence.

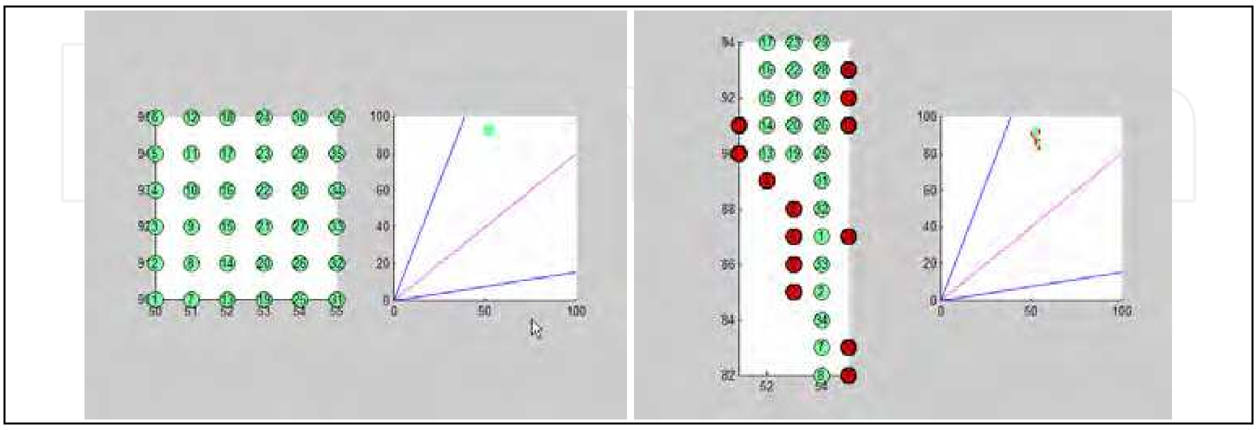


Figure 16.(a) Simulation with a double potential and perception of de state

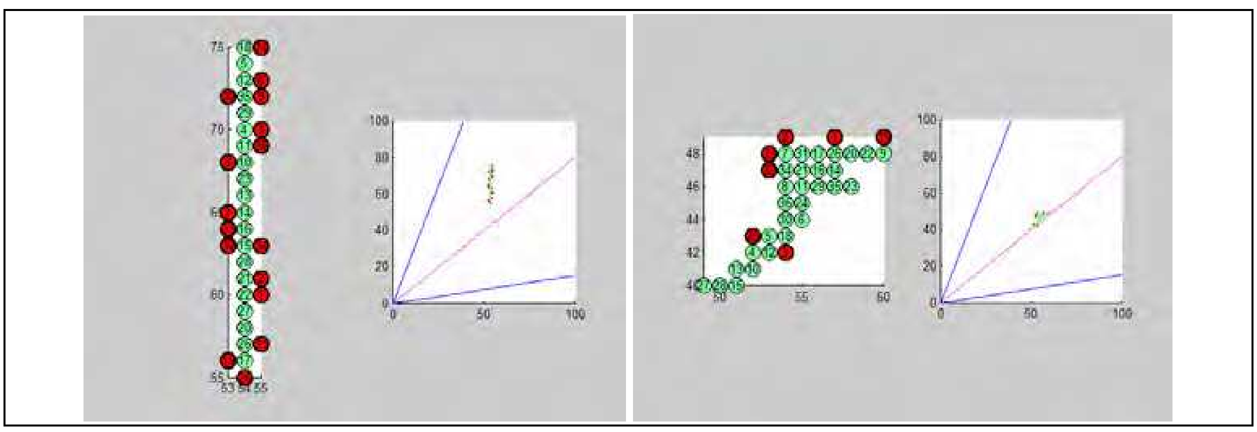


Figure 16.(b) Simulation with a double potential and perception of de state

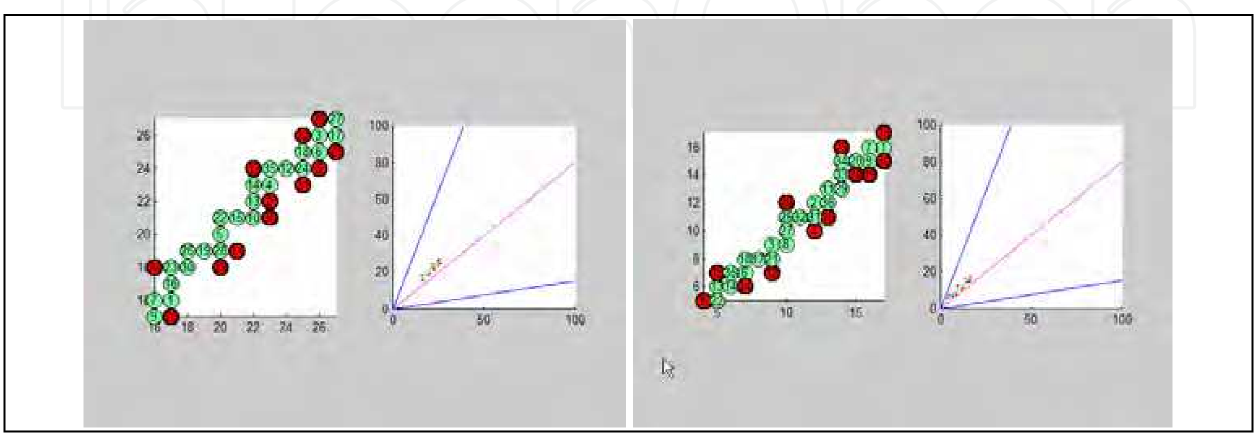


Figure 16.(c) Simulation with a double potential and perception of de state

8. Conclusion

This study corresponds to the calculation of a high level decision of the choice the modular self-reconfigurable robots destination. As a result in experimental studies one can see that any little difference in reactive algorithm has emergent characteristics.

The algorithms with state perception bring the desired emergent behaviour witch is to propose an emergent solution for the problem of displacement, using a distributed approach based on the cooperation of the modular robots. The collective displacement to reach the target by self-reconfiguration emerges with random scheduling as a process of individual and modular behaviour.

We propose a minimal communication system for the knowledge of the state of the robot "active or not", to avoid the problem of dynamics blocking patterns. We present a potential field with two referential measures, a set of properties, two lemmas and a theorem that guarantees the convergence of the emergent solution.

9. References

Amblard F. & Phan D. (2006). Modélisation et simulation multi-agents, applications pour les Sciences de l'Homme et de la Société. Lavoisier, Hermes science, Paris 2006.

Arkin R. (1998). *Behavior-based robotics*, The MIT Press, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1998.

Carrillo E. & Duhaut D. (2005). Methods for collective displacement of modular self-reconfigurable robots. *Proceedings of the CLAWAR05 Conference*, London, UK. 2005.

Christensen D. (2005). HYDRA From Cellular Biology to Shape-Changing Artefacts, *Lecture Notes in Computer Science*. Springer Berlin, Heidelberg, September 2005.

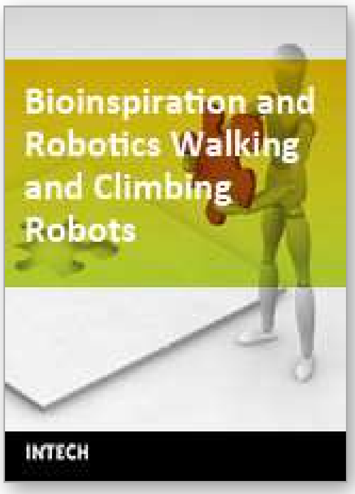
Duhaut D. & Carrillo E. (2007). Study of Emergence on Multi-Robot Applications, *Robotics and Automation Conference*, March 26-27. 2007.

Fredslund, J.; Mataric M.(2002). A General, Local Algorithm for Robot Formations, *Transactions on Robotics and Automation, special issue on Advances in Multi-Robot Systems*, IEEE, 18 October 2002.

Yoshida, E.; Murata, S.; Kamimura, A.; Tomita, K.; Kurokawa, H.; & Kokaji S. (2001) A Motion Planning Method for a Self-Reconfigurable Modular Robot, *Proceedings of the International Conference on Intelligent Robots and Systems* 2001 IEEE/RSJ.

MAAM, (2007). Internet address: www-valoria.univ-ubs.fr/Dominique.Duhaut/maam/publications.htm

Reynolds C. (2007). Internet address: http://www.red3d.com/cwr/



Bioinspiration and Robotics Walking and Climbing Robots
Edited by Maki K. Habib

ISBN 978-3-902613-15-8
Hard cover, 544 pages
Publisher I-Tech Education and Publishing
Published online 01, September, 2007
Published in print edition September, 2007