Mechatronics 19 (2009) 463–470

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

A cooperative multi-robot architecture for moving a paralyzed robot

Olivier Simonin [a],*, Olivier Grunder [b]

[a] LORIA Laboratory, INRIA Maia Project and Université Henri Poincaré, Campus Scientifique-BP 239, 54506 Vandoeuvre-lès-Nancy Cedex, France
[b] SeT Laboratory, Universite de Technologie de Belfort-Montbeliard, 90010 Belfort Cedex, France

ARTICLE INFO

Keywords:
Autonomous mobile robots
Multi-agent systems
Cooperation
Self-adaptive control
Artificial intelligence

ABSTRACT

In this paper we study the possibility of cooperation between reactive and deliberative based robots, which are generally considered as antinomic approaches. We focus on a case study composed, on the one hand, of an “intelligent” robot that submits failures which prevent it from moving, and on the other hand, of a pool of simple autonomous mobile robots which are able to push. The paralyzed robot can broadcast signals to recruit mobile robots and to be pushed by them. These signals are interpreted as force fields by agents in order to compute their reactive behavior. We present these different robot behaviors and analyse two experiments. We show that the proposed system provides a control loop which is independent of the number of robots pushing on each arm, showing that a combination of multi-agent and deliberative architectures can define intelligent and robust multi-robot systems.

© 2008 Elsevier Ltd. All rights reserved.

1. Introduction

In future factories or hostile environments (like nuclear plants), robots will be partially or totally autonomous, and they will have to work together to carry out tasks. Under extreme conditions, failures may occur and robots will have to cooperate, or reconfigure, to help each other [14]. This paper focuses on this kind of cooperation where simple mobile autonomous robots help another kind of robot.

For this purpose, we study how the reactive multi-agent approach [5] can be applied to a pool of mobile robots that have to cooperate in order to help a robot that has different abilities. This paper aims at exploring such an approach to define a model for cooperation between heterogeneous robots. In particular, we show that combining behavior-based robots and deliberative ones can define intelligent and robust multi-robot systems.

Reactive architectures rely on the perception-reaction principle [4]. Such agents or robots have neither direct communication, i.e. they do not use high level communications, nor environmental representation [2]. Cooperation between agents is based only on bio-inspired mechanisms, such as direct perception, signal broadcasting, or indirect communication via environment changes [13,3,8,7]. On the other hand, deliberative architectures rely on high level communications, rich sensors and a representation of the environment which allows for planning actions [6,9,1]. In this paper, we propose a model to study the interaction between these two approaches, generally considered as antinomic.

The system under consideration focuses on transportation tasks, where mobile robots must cooperate to transport other ones. In particular, this system is composed first of an “intelligent” (deliberative) robot that submits failures preventing it to move, and second, of a set of reactive-based mobile robots only able to capture signals and to move individually. The paralyzed robot has to reach a certain goal point that can be, for example, an emergency exit or a repair area. It is able to send signals to attract the mobile robots so that they will push it in a specific direction. So, this system presents two types of cooperation. An indirect cooperation between the mobile robots to perform a coordinated pushing action and a direct one, i.e. using communication, between the deliberative robot and the simple ones. The whole system can be seen as a box-pushing task where the box is a robot asking to be moved along a specific path.

The remainder of the paper is organized as follows: Section 2 introduces the robotics system under consideration and the objectives of the cooperation. Then Section 3 presents the deliberative and reactive behaviors studied to achieve the task. In Section 4 we describe two experiments with real robots and then analyze the results. Section 5 discusses the two kind of cooperation involved in the proposed model. Section 6 gives a conclusion of this study and discusses future work.

2. Problem description

The key idea of the proposed architecture is to use only simple behavior to perform a complex task. In particular, robot communications are limited to broadcasting simple signals. The perceptions of mobile robots are also limited as they can only perceive their local environment and the emitted signals.

---

* Corresponding author. Tel.: +33 (0)3 83 59 30 43; fax: +33 (0)3 83 27 83 19.
E-mail addresses: olivier.simonin@loria.fr (O. Simonin), olivier.grunder@utbm.fr (O. Grunder).

0957-4158/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mechatronics.2008.11.006

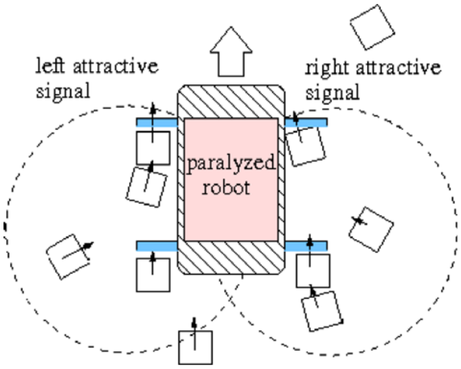

Fig. 1. Principle of the cooperative task.

Fig. 1 presents the principle of the system being studied. A robot, considered paralyzed due to a failure or without the ability to move, has to reach a specific destination, for example an emergency exit. To achieve this objective, it is placed on a carriage to facilitate its displacement, and will be pushed by a set of simple mobile robots that evolve in this environment. These robots are only able to perceive close obstacles and to react to signals. The objective for the paralyzed robot is to recruit these simple robots so that they push its carriage and move it along a desired path. To achieve such a goal, we assume that it can emit attractive signals from both of its sides, called left and right signals. Each signal covers a limited circular area (see Fig. 1). In contrast to the mobile robots, the paralyzed one has a vision of the environment to determine the path it wants to follow. As this robot and its carriage are considered heavy, the displacement can only be carried out if several mobile robots push it simultaneously. Note that there has to be cooperation not only between the two different types of robots, but also between the mobile ones.

3. Methods

This section presents the different behaviors for both kind of agents, deliberative for the paralyzed robot, and reactive for the mobile robots.

3.1. The paralyzed robot

The main goal of this agent is to reach a destination point following a path that can be determined in advance, or perceived during the progression of the robot. As it now cannot move, the behavior of this agent is limited to requesting help by emitting signals. However, to be pushed in a particular direction, the agent can control its requests by emitting signals from both of its sides. To perform this task the agent has

- an environment perception system that is used to determine path error,
- two signal emitters, one for each side of the carriage, which are used, first, to attract mobile robots, and second, to transmit to them the relative forces needed to move in the right direction.

The intensity of the signals is in direct relation with the rotation or translation needed by the agent to move in the desired direction: the stronger the signal, the stronger the pushing force will be. Consequently, the intensity of each signal changes dynamically as the robot is moved along. We detail in Section 4.3.1 the computation of the signal intensities used in the experiments. Fig. 2 shows a flowchart of the paralyzed agent's behavior.

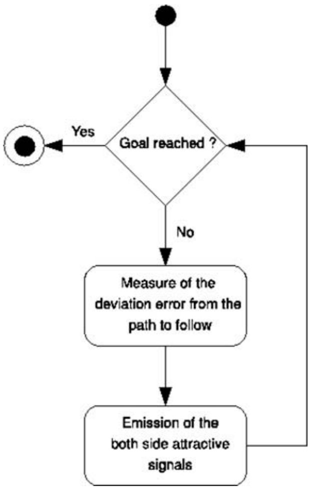

Fig. 2. Behavior of the paralyzed agent (placed on a carriage).

3.2. The pusher robots: a force field approach

These robots explore the environment until they perceive attractive signals emitted by the paralyzed one. Their behavior then consists of moving towards the origin of the signal, whose direction can be easily computed from the set of sensors that encircle the robot's body (see Fig. 3). As a consequence, they will arrive near the carriage and may collide with it in order to apply a pushing force. However, this task needs to be precise so that the agents are placed correctly and push against the arms of the carriage (see Fig. 1). For this purpose two simple reactive behaviors are defined: (i) attraction towards the carriage by following the signal (see Fig. 4a), (ii) sliding along the carriage sides (see Fig. 4b). The objective is to point the arms in a direction so they can be pushed (see Fig. 4d). We detail now these fields and how they are used.

- If the robot is inside one of the two signal-quadrants, it can deduce the signal direction, which is oriented towards the point p_0 situated at the front of the carriage (see Fig. 4a). This direction can be approximated from signal receptors which are activated on the pusher robot.
- A sliding force is considered when the pusher is close to the carriage (i.e. in the sliding area, see Fig. 4b). The sliding direction is perpendicular to the carriage side (see below).
- An avoiding force from other mobile robots in proximity is integrated only outside the sliding area.

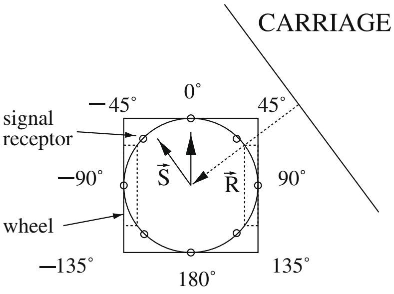

Fig. 3. Sliding force $\vec{S}$ computed from the perception of $\vec{R}$ vector.

Three force vectors can be derived from these perceptions:

- An attraction force towards the signal source $\vec{A} = p\vec{p}_0$. The intensity of the signal reception allows to approximate the norm $\| \vec{pp_0} \|$. Note that the attraction force $\vec{A}$ decreases with the distance to p_0 and is bounded to a maximum value. $\|\vec{A}\| = max(\|p\vec{p}_0\|, A_{max})$.
- A sliding force $\vec{S}$ along the carriage such that $\vec{S}.\vec{R} = 0$, where $\vec{R}$ is the vector from the closest point of the carriage to the robot (see Fig. 3). The intensity of $\|\vec{S}\| = R_{max} - \|\vec{R}\|$, which increases when the robot goes near the carriage. The direction of $\vec{S}$ is given by the perceived signal. Considering $\vec{R}$ as reference, when the robot is inside the left signal, $\vec{S}$ is oriented on the right-hand side, or else on the left-hand side (see Fig. 4b).
- A force $\vec{D}$ to avoid collisions with other nearby robots. A sliding force is computed for each of them. $\vec{D}$ is a weighted sum of these vectors as a function of their respective distance, see details in [15].

The selection and combination of the forces are carried out following a subsumption architecture [4] giving the current direction $\vec{V}$:

If ARM PERCEIVED then

$$\vec{V} = g_5\vec{j}$$

If SIGNAL DETECTED AND CARRIAGE PERCEIVED then

$$\vec{V} = g_3\vec{A} + g_4\vec{S}$$

If SIGNAL DETECTED AND NO CARRIAGE PERCEIVED then

$$\vec{V} = g_1\vec{A} + g_2\vec{D}$$

Otherwise

$$\vec{V} = g_1\, \overrightarrow{Random}(\vec{V}) + g_2\vec{D}$$

Here, g_1 normalizes the vector direction of the robot when moving randomly, and g_2 allows to weight the obstacle/robot avoiding force $\vec{D}$ when combined with $\vec{V}$. We generally set $g_2 = 2g_1$. The same gains are used between the attraction force towards the signal's origin and the avoidance of other robots. When the robot enters the sliding area, the g_3 and g_4 gains weight the sliding force $\vec{S}$ and the signal attraction $\vec{A}$ (g_4 is chosen slightly superior to g_3 so that the pusher avoids collision with the carriage). Finally, g_5 is the gain for the pushing force on arms, when they are perceived, and is a function of the signal information (see details in Section 4).

The agents are equipped with sensors for detecting arms (for instance, a simple color detection can be used) so as to be attracted by them (see Fig. 4c). So, when an agent detects an arm of the carriage, it just has to move forward very close to the arm and then push it using a force proportional to the signal attraction.

By following this process the pusher robots will be oriented towards the arm sides of the carriage, as represented in Fig. 4d, and will push on the arms when they are perceived. This combination of influences is efficient (see next section), however, no more than one pusher can work on an arm. To allow several agents to combine their forces on a same arm, the back of the pusher robots has been colored in the color arm. As any agent perceiving the arm color moves forward to exert a pushing force, it will push either against an arm or another pusher. This whole behavior is presented in Fig. 5.

4. Experimental results

To experiment the proposed architecture, two different types of robots were chosen. The first one being the "intelligent" robot would be prevented from using its moving actuators, while still being able to take decisions and to send signals to mobile robots. A Lego Mindstorms robot was chosen to be the paralyzed agent (see Fig. 6a). The second type of robot, the pusher, needed to be a very basic one that could just move and push small items, but with no embedded intelligence. For this, Mirosot soccer robots were used (see Fig. 6b).

4.1. Hardware

4.1.1. Mirosot

The Mirosot robot is cubic with a side length of approximately 7.5 cm. Movement is controlled by adjusting the speed of each

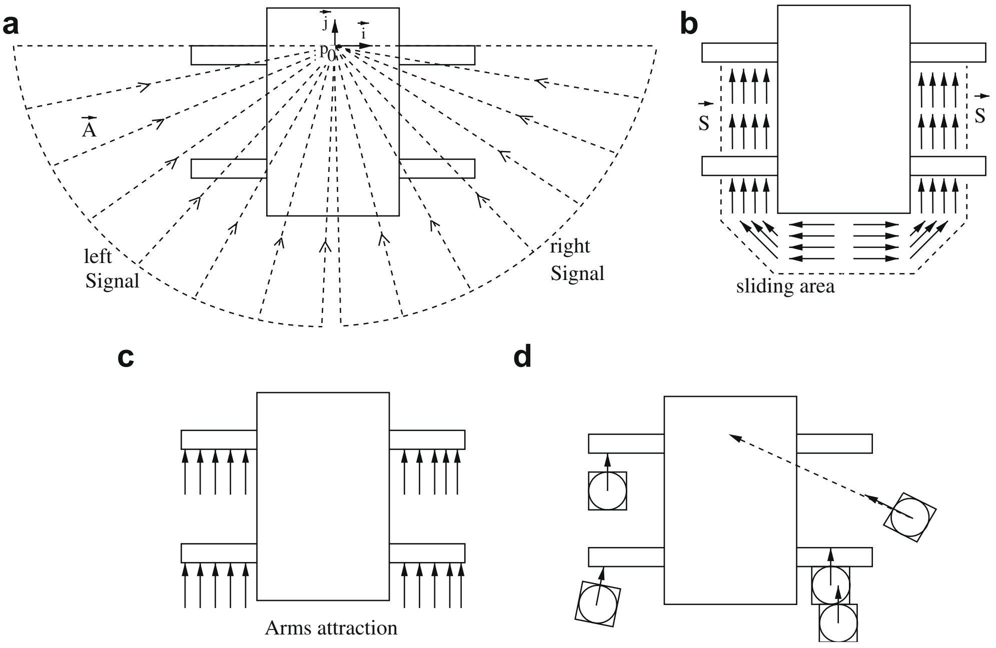

Fig. 4. Illustration of the Force fields used to define the pushers' motion: (a) signals of attraction towards the top of the carriage, (b) sliding field along the carriage from obstacle perception, (c) pushing forces when arms are perceived and (d) examples of robots inside the two signals, showing their current motion direction.

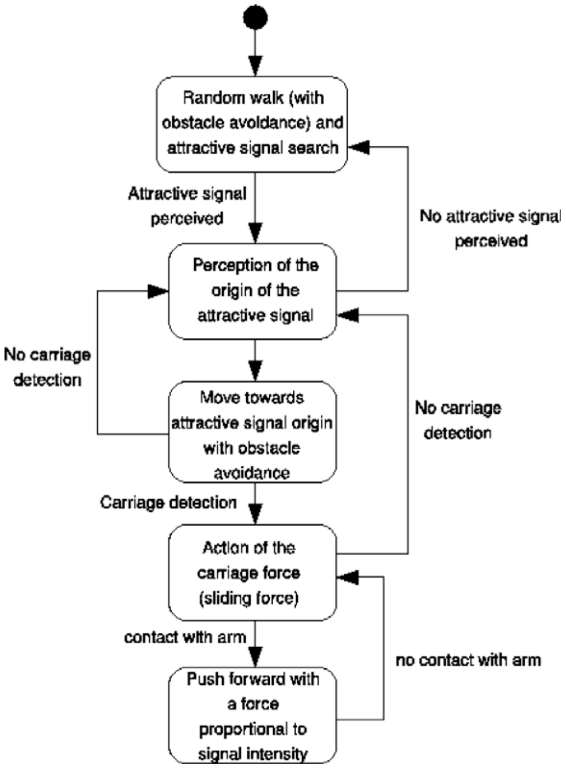

Fig. 5. Behavior of the pusher agents.

wheel. Control is achieved only from the host computer setting speed values for the left and right wheels of each robot. Communication is achieved via an RF system. The host system computes the location and the orientation of each robot from the camera placed over the ground (see Fig. 7). Due to its simplicity, such a system is well suited to play the role of a reactive pusher robot.

4.1.2. Mindstorms

The Mindstorms robot is based on a RCX brick, which is a Hitachi H8 programmable microcontroller connected to three motors, three sensors, and an infrared serial communications interface. The microcontroller is composed of an external RAM of 32K and an on-chip ROM 16K that contains the drivers to address the RCX devices. The RAM of the RCX is shared between the firmware and the user programs that can be downloaded as byte code.

4.1.3. The carriage

As in the experiment the Mindstorms robot is considered to not be able to move using its own actuators, it was placed on a carriage without motors. It was, however, equipped with four arms which were long enough to allow a Mirosot soccer robot to apply a force on it (Fig. 8).

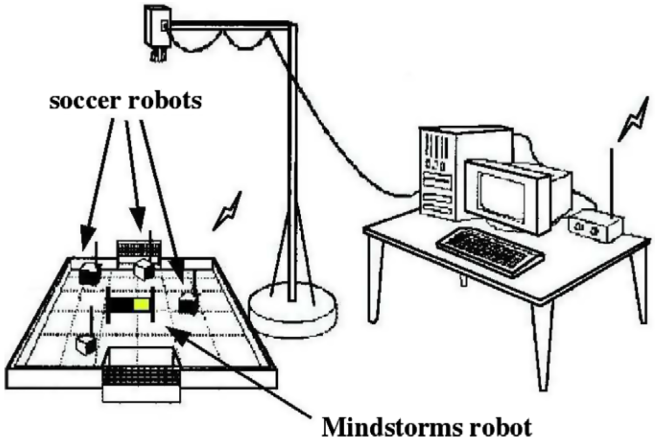

Fig. 7. Diagram of the multi-robot system.

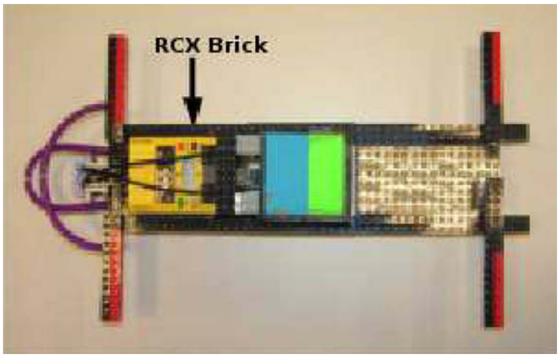

Fig. 8. Carriage top view.

4.2. Communication

The communication between the two kinds of robots is a one-way communication: the paralyzed one has to transmit its orders to the mobile ones. The main problem with this communication resides in the heterogenous devices used for this purpose: the Mindstorms robot uses infra-red communication and the Mirosot ones have to be driven using a hertzian wave transmitter.

The communication between the computer and the RCX was achieved by the means of an infra-red transceiver.

To simulate the direct interactions between the Mindstorms robot and the reactive ones, messages are sent from the RCX brick (playing the role of the paralyzed robot) to the Mirosot mobile robots via the controlling PC. On the one hand, the PC receives the values of the two attractive signals from the infra-red transceiver. On the other hand, it computes the possible reception of signals by the Mirosot robots by considering their position.

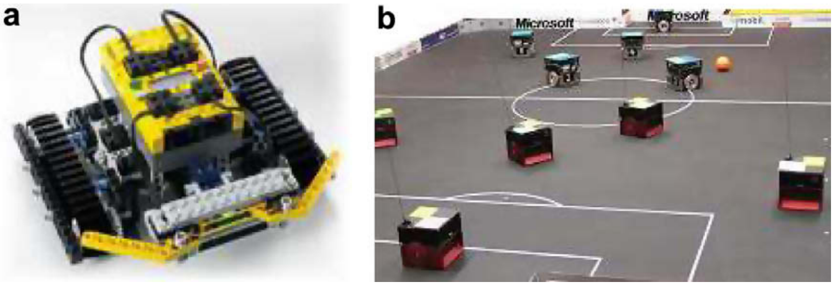

Fig. 6. (a) Lego Mindstorms robot and (b) Mirosot robot soccer.

In this experiment the information transmission corresponds to the following scheme: the Mindstorms robot has to follow a path delineated by a white line, using two light sensors. Depending on the light variations of these sensors, the Mindstorms has to decide which direction is more appropriate so that it could follow the path defined by the line. The Mindstorms converts the direction information into attractive signal values and sent them to the soccer PC computer using the Mindstorms IR communication protocol.

4.3. Description of the experiments

This section presents the results of the experiment through two experimental setups. In both setups a paralyzed Mindstorms robot must attract mobile robots to be oriented and pushed along a white line drawn on the floor by them. The first setup is a validation of the proposed principle, using a fixed number of pushers. The second experiment is more complex as it involves a varying number of pushers and taking a right-angle bend. We first give the expression of the attractive signals as well as the values of the other gain parameters, afterwards detailing both experiments and their results.

4.3.1. Expression of the attractive signals

When an arm of the carriage is perceived by a pusher robot, it will apply a force on this arm corresponding to the following expression: $\vec{V} = g_5\vec{j}$. We now define the gain g_5 that is proportional to the attractive signal emitted by the Mindstorms.

Let l_{right} and l_{left} be the values perceived by the right and left light sensors, and L_{max} the maximal value available for l_{right} and l_{left}. We note Δl the difference between l_{left} and l_{right}: $\Delta l = l_{right} - l_{left}$. The difference between the right and the left light sensor values is used to compute the intensity of the signals emitted by the Mindstorms on its right side i_{right} and on its left side i_{left}, by using the linear representation of the system (1).

$$
\begin{aligned}
i_{right} &= I_{max}.\frac{L_{max} - \Delta l}{2.L_{max}} \\
i_{left} &= I_{max}.\frac{L_{max} + \Delta l}{2.L_{max}}
\end{aligned}
\qquad (1)
$$

where I_{max} is the maximal value that can be emitted by the Mindstorms on any of its sides.

When the carriage follows the right path, both sensors are on the white line, and thus detect a high intensity of light with roughly the same values. Consequently, $\Delta l = 0$ and $i_{right} = i_{left} = I_{max}/2$, which means that the carriage requires to be pushed with the same force from the right arms and from the left arms, to move forward.

Conversely, when the carriage moves off the line from the right for example, the brightness measured by the right sensor decreases. As a consequence, the expression Δl decreases and tends toward $-L_{max}$, because the left sensor is still on the white line, and gives a value close to L_{max}. On the right side, the intensity of the attractive signal becomes stronger, tending toward I_{max}, and on the left side the intensity of the attractive signals decreases to zero. As we can observe, if these forces are really applied to the carriage, it will turn left and go back onto the line.

In our experiments, Mindstorms robots set L_{max} equal to 100 as the light measure is a percentage. For the communications, we set I_{max} to 0.1 which corresponds to the maximum robot speed we want 0.1 m s^{-1}.

The gain g_5 corresponds to the intensity of the perceived signal i_{right} (resp. i_{left}) if the pusher robot is on the right (resp. the left) of the carriage. In practice g_5 is multiplied by a coefficient, which fine-tunes the forces.

The other gains used in the force combinations are the same in both experiments, and are as follows: $g_1 = 1.0$, $g_2 = 2.0$, $g_3 = 1.0$ and $g_4 = 1.4$.

4.3.2. Experiment #1

The setup of the first experiment is as follows:

- Objective: a Mindstorms robot, placed on a carriage, has to be oriented and pushed along a short white line. The carriage is initially misdirected, see Fig. 9.
- Pusher robots: three mirosot robots wander initially around the carriage.

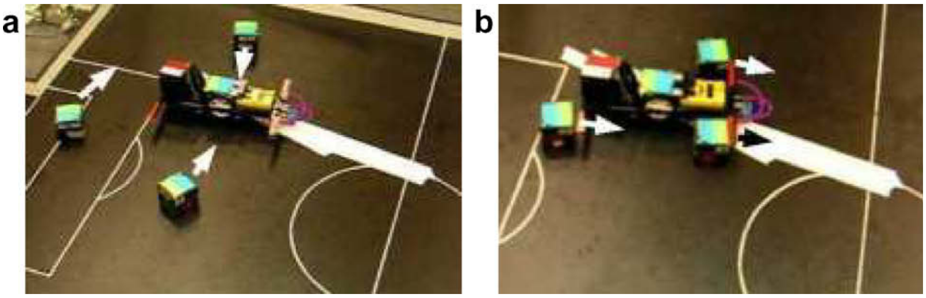

Fig. 9. (a) Placement phase: the Mindstorms attracts the soccer robots, (b) pushing phase: the three robots push on the arms of the carriage.

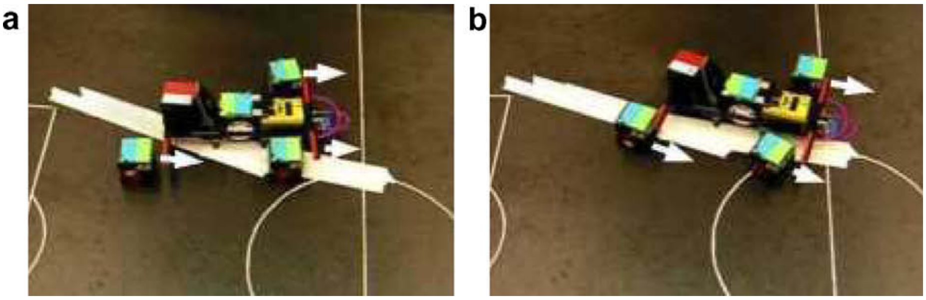

Fig. 10. (a) The carriage goes a wrong way, (b) the carriage has been rotated due to the force applied on its left side.

We used snapshots from a video to analyse the different phases of the first experiment (Fig. 9a to Fig. 10b).

The first step consists of the positioning phase. Mobile robots perform a random walk until they perceive attractive signals (this is the case for the three robots in Fig. 9a). They then start to move towards the carriage. The obstacle avoidance procedure ensures that no collision occurs between the mobile robots. While approaching the carriage the robots are braken due to the perception of the carriage as an obstacle. When entering the sliding area, the combination of attractive and sliding forces tends to align the robots along the carriage (if they are placed between two arms). Moreover when pusher robots detect an arm they change their behavior to simply move closer to it (see Fig. 9b). They then start to push the carriage.

The second step consists of mobile robots pushing, using a force in proportion to the perceived signal. Consequently, the Mindstorms robot has to evaluate its displacement to define its two signals. Roughly, we can say that the more the Mindstorms has to turn, the higher the difference between the right and the left emitted signals has to be.

Fig. 10a shows that the three robots have pushed the carriage but without following the white line. This can be explained by the fact that there are two robots on the right side of the carriage, while there is only one on the left side. Thus, the Mindstorms robot sends a strong attractive signal on its left side and almost nothing on its right side. As a consequence the robot on the left side increases its applied force, which rotates the carriage while moving it. During this moving, the Mindstorms perceives when its direction corresponds to that of the white line, and sends two signals having an intensity which is inversely proportional to the number of pushers on each side of the carriage (see Fig. 10b). The system tends to balance the pushing forces to go forward along the line.

4.3.3. Experiment #2

The second experiment consists of following a longer line with a corner to take, and a variation in the number of mobile robots:

- Objective: a Mindstorms robot, placed on a carriage, has to be pushed along the white line until it has passed the corner, see Fig. 11.
- Pusher robots: first, three Mirosot robots wander about, then a fourth was added afterwards.

Figs. 11–13 show six snapshots of experiment number 2, corresponding to the video http://www.loria.fr/~simoniol/FourPushers.avi [12]. Watching the video allows for a better analysis of the different behaviors and stages. In particular, it shows how the system self-adapts when a fourth pusher robot arrives during the displacement carried out by the initial three (see Fig. 12).

Different experiments have shown that reactive mobile robots move and respond to Mindstorms's signals quickly. However

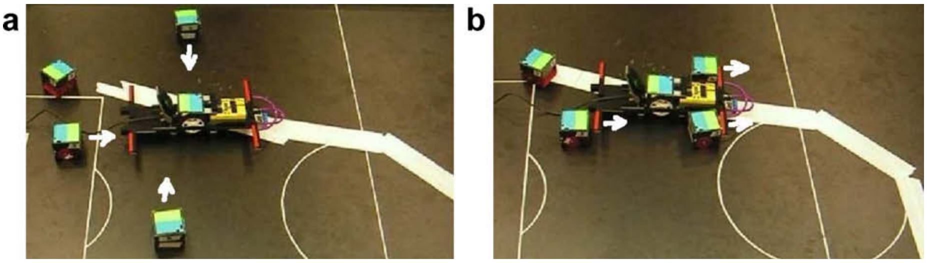

Fig. 11. (a) Placement phase and (b) pushing phase with three mobile robots.

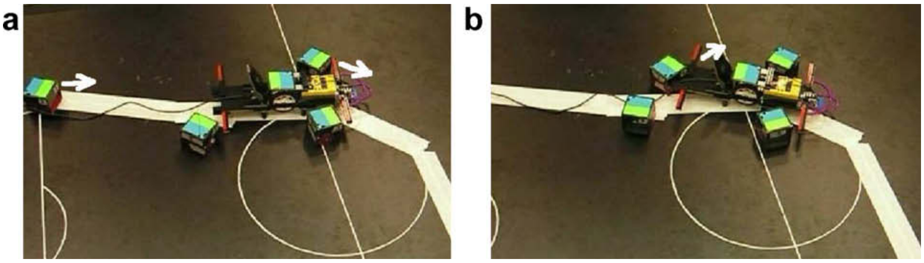

Fig. 12. (a) Introduction of a new robot and (b) pushing action of the new robot on the carriage.

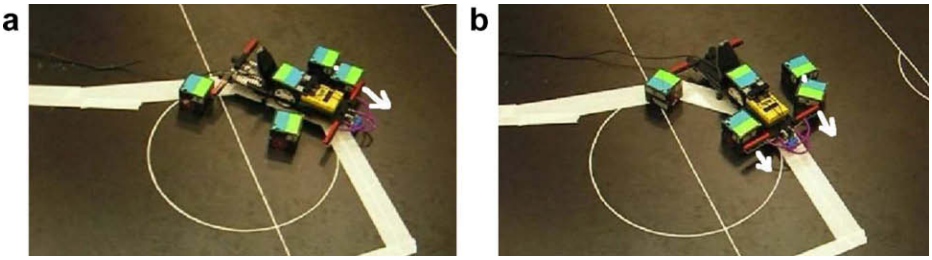

Fig. 13. (a) Replacement of the new robot and (b) combined action of the left pushers on the carriage.

obstacle avoidance and pushing actions require moving at low speed (in experiments, the maximum speed of the pushing action is 0.1 m s^{-1}). The speed that the carriage moves is then slightly under that of the pushing robots' due to the rotation and approach phases.

The aim of these experiments was to focus on self-organization when the number of pusher robots varies. We conducted experiments involving two to five mobile robots (the maximum available to us). We observed that the growing the number of robots allows for a more precise control of the carriage. Adding the force of several robots' gave a smoother movement than, for instance the binary reaction of one pusher per side. As shown in Fig. 12, the number of pusher robots can vary during the displacement without compromising its completion. This ability results from the perception-based control which relies only on the measured direction error. Note, however, that numerous changes in the number of pushers can have a negative impact on the carriage movement, and can lead to the impossibility of moving in the correct direction. This self-adaptation, or re-configuration, is then efficient when there are not too many changes. In particular it allows to define a robust solution in case of mobile robots failures.

5. Discussion

Numerous works have shown that the reactive approach is efficient for performing a cooperative box-pushing task [7]. In this paper we extended such a task by replacing the box with a paralyzed robot able to interact with robot pushers. We now discuss the different kind of cooperation involved in the proposed model.

There is, first, a direct cooperation between the paralyzed robot and the mobile ones. In this case, the mobile robots only answer to the requests coming from the Mindstorms robot. This one-way cooperation can be characterized as "altruistic" because the mobile robots act only to help the others, not for their own benefit (see such a concept in [11]). However, supposing the simple mobile robots aims at reaching the same goal as the Mindstorms robot, they find a solution by helping it. So, since there is also a direct benefit for them, there is a strong cooperation between the robots. Such a task is presented in [10] as cooperation between a blind robot and a paralitic one, which defines an efficient system.

A second type of cooperation appears between the pusher robots. They have to simultaneously push the carriage so that the force is strong enough to move it. In our approach there is no explicit coordination for performing this task. However, it can be observed that several robots combine their forces and try to push on the arms until they are numerous enough and strong enough to move the carriage. Such a result relies on the fact that each mobile agent has the same reactive behavior. This reactive behavior directly links the agent's perceptions to its pushing actions. More forces are naturally added onto the carriage, defining a collective action. Signals ensure that these forces are well distributed on both sides of the carriage. This is an example of an indirect cooperation because no direct communication is necessary to combine the mobile robot's actions.

The pushing action from one mobile robot to another has been validated in simulation and with real robots (shown in Fig. 13b). When such a robot perceives the arm color on the body of another robot, which is situated close to the carriage, it moves towards this color and consequently pushes the corresponding robot. More forces are naturally added and the carriage can be moved or rotated more quickly.

A major property of the proposed system is to provide a control loop which is independent of the number of robots pushing on each arm. The paralyzed robot just has to adapt the signals intensity following the measure of its navigational error. Finally, the mobile robots constitute a robust actuator for the paralyzed robot.

6. Conclusion

In this paper, a heterogeneous robotics architecture, allowing for cooperation between reactive and deliberative agents, has been presented. The task that was studied can be seen as box-pushing problem, where the box is replaced by a paralyzed robot asking to be moved along a specific path. Mobile robots that evolve in the environment can push the paralyzed robot by using a perception and combination of force fields.

It has been shown that through the cooperation between the mobile robots, the paralyzed robot can move as if it was equipped with actuators. There is direct cooperation between both types of robots in order to move the whole system along a specific path. Moreover, indirect cooperation appears between mobile robots that have to add their forces to move the paralyzed robot.

Experiments have been presented, and validate the approach. The proposed architecture is robust and can self-adapt to changes or disturbances in the system (e.g. adding or removing pusher robots during transportation). This results from using a reactive multi-agent approach to coordinate pusher robots and enabling a direct link between the measurement of the error variation and the force that the pushers must apply on each side of the paralyzed carriage.

Future research projects are two fold. On the one hand, we plan to make the global system more autonomous by removing all communication bridges and navigation aid subsystems (e.g. positioning, obstacle detection). To achieve this objective, we will apply our architecture to other type of robots, in particular to more autonomous and communicative ones, like e-puck and NXT Mindstorms robots. The main advantages of this choice are based first, on a common way of communicating (Bluetooth), and second, on a set of inherent mechanisms for carrying out the various tasks related to navigating with the considered system (e.g. obstacle detection, environment perception). On the other hand, we will further study the influence of the number of pusher robots on the quality of the resulting control. In particular, we intend to determine the optimal number of pushers and the maximum limit for efficient functionning. Finally, we plan to evaluate this type of architecture on other cooperative tasks involving mobile robots. Another very interesting application is the assistance that can be brought by mobile robots to human beings that need to be rescued.

References

[1] Alami R, Botelho S. Plan-based multi-robot cooperation in plan-based control of robotics agents. Lect Notes Comput Sci 2002;2466.

[2] Arkin R. Behavior based robotics. The MIT Press; 1998.

[3] Beckers R, Holland O, Deneubourg J-L. From local actions to global tasks: stigmergy and collective robotics. In: Artificial life IV: proceedings of the fourth international workshop on the synthesis and the simulation of living systems. MIT Press; 1994. p. 181–9.

[4] Brooks R. A robust layered control system for a mobile robot. IEEE J Robot Automat 1986;RA-2(1):16–23.

[5] Ferber J. Multi-agent systems an introduction to distributed artificial intelligence. Addison-Wesley; 1999.

[6] Ghallab M, Nau D, Traverso D. Automated planning, theory and practice. Elsevier, Morgan Kaufmann Publishers; 2004, ISBN 1-55860-856-7. 663 pages.

[7] Kube C, Parker C, Wang T, Zhang H. Biologically inspired collective robotics. In: Recent developments in biologically inspired computing; 2005 [chapter 15].

[8] Mataric M. Behavior-based control: examples from navigation, learning, and group behavior. J Exp Theoret Art Intell 1997;9(2–3):323–36.

[9] L. Parker, Current state of the art in distributed autonomous mobile robotics. In: Parker LE, Bekey G, Barhen J, editors. Distributed autonomous robotic systems, vol. 4; 2000. p. 3–12.

[10] Rongier P, Liegeois A, Simonin O. Markovian analysis of a heterogeneous system: application to a cooperation task for multiple consumer robots. In: SMC'2000, Knoxville, TN, USA; 2000. p. 3033–8.

[11] Simonin O, Ferber J. Modeling self satisfaction and altruism to handle action selection and reactive cooperation. In: Sixth international conference on the simulation of adaptive behavior (SAB 2000), vol. 2; 2000. p. 314–23.

[12] Simonin O, Grunder O. http://www.loria.fr/~simoniol/FourPushers.avi; 2008.

[13] Steels L. Cooperation between distributed agents through self-organization. In: Decentralized AI – proceedings of the first European workshop on modelling autonomous agents in a multi-agelit world (MAAMAW-89). Amsterdam: Elsevier Science BV; 1989. p. 175–96.

[14] Yim M, Roufas K, Duff D, Zhang Y, Eldershaw C, Homans S. Modular reconfigurable robots in space applications. Autonom Rob J 2003;14(2): 225–37 [special issue for Robots in Space].

[15] Zeghal K. A comparison of different approaches based on force fields for coordination among multiple mobiles. In: Proceedings of the 1998 IEEE international conference on intelligent robots and systems, Victoria, BC, Canada; 1998.