Evaluation of a Fault-tolerant Model for Tactic Operations of Mobile Robotic Groups Using Genetic Algorithms

Gustavo Pessin
University of São Paulo (USP)
Institute of Mathematics and Computer Science (ICMC)
Av. Trabalhador São-carlense, 400 – C.P. 668 – 13.560-970 – São Carlos, SP
pessin@icmc.usp.br

and

Fernando Osório
University of São Paulo (USP)
Institute of Mathematics and Computer Science (ICMC)
Av. Trabalhador São-carlense, 400 – C.P. 668 – 13.560-970 – São Carlos, SP
fosorio@icmc.usp.br

Abstract

This paper addresses the evaluation of a fault-tolerant model for tactic operations of mobile robotic groups. The coordinated action of the group is planned with Genetic Algorithms (GA) and aims to act on an environmental disaster scenario, simulated as a forest fire. The robotic squad should surround the fire and avoid fire's propagation. Initially, we evaluate several parameters in the GA, seeking to obtain the set of parameters that would accomplish the more efficient evolution. Then, we simulate failures in robots operations in order to evaluate strategies of reorganization. The simulation's results[1] showed that with an adequate set of parameters it is possible to get satisfactory strategic positions to coordinate and to reorganize the robotic group in case of robot failures.

Keywords: Genetic Algorithm, Multi-robot systems, Coordination.

INTRODUCTION

The continuous evolution provided by mobile robotics research area has made even more efficient robots for several functions. Research about controlling complex motor functions are developed on several research centers around the world, encompassing studies about sensors and actuators, positioning, navigation and localization in addition to many other requirements related to robotic hardware, as demonstrated by [1] and [2]. The development of specialized algorithms composed by rule based systems and finite state automata have been done in order to coordinate these physical sets in a dynamic environment, representing a complex challenge [3]. Giving to autonomous mobile robots the ability of intelligent reasoning and ways to interact with the environment is a research challenge which has attracted the attention of a great number of researchers [1].

One of the primary goals of robotic systems usage is that they can help in tasks extremely dangerous to human beings, like cleaning nuclear residuals, chemical accidents, fire combat, search and rescue in dangerous environments or even on constructions, agriculture, hostile environment exploration, security and critical missions. There are many fields where a single robotic agent is not enough to perform a task, and in several activities a better idea is to use multi-robotic systems. Multi-robotic systems are systems where autonomous mobile robots work cooperatively to complete a mission with direct or indirect robots interaction [4]. These systems are extremely dependent on control techniques. Related to the applicability, multi-robotic systems can add mobility, flexibility and robustness of a new way to a wide range of new applications [5]. Monitoring and combat of forest fire is an example of multi-robotic system that could considerably reduce human, material and environmental losses.

In the firefighting mission, one of the most important questions is related to the robot trajectory and final destination position settings. According to the actuation capability of each robot, weather conditions (wind, rain), topography, and vegetation, several arrangements could be proposed in order to surround and put out the fire. These

---

arrangements, when suggested by a specialist, should take in account a large number of variables, which makes it a hard task. In these cases, machine learning techniques may be successfully used. One of the machine learning technique which has been showing satisfactory results in solving optimization problems is Genetic Algorithms [6], [7], which consists in a global optimization algorithm that employs parallel and structured search strategies, directed by fitness points seeking. Allowing the multi-criteria search in a multidimensional space and due to be unsupervised, it does not need any previous information or solution database.

In [8] we proposed a Genetic Algorithm to accomplish the formation of a robotic squad that should perform a firefighting task; the previously proposed GA did not deal with reorganization in case of robot failures. So, if any robot suffered some damage or got stuck, the operation might not be satisfactorily performed. In this paper we present a reformulation of that work, seeking to make possible perform the robotic squad firefighting task even with robotic failures. Considering that convergence speed is one of the most important aspects for the proposed system (to get new robot positions to perform the task), we evaluate several parameters in the GA, seeking to obtain the set of parameters that would accomplishs the quicker and more efficient evolution; and then, we simulate failures in robots operations in order to evaluate strategies of reorganization.

This paper has the following structure: Section 2 introduces short theoretical description of robot's applications. Section 3 introduces a brief theoretical description of Genetic Algorithms and its application to robotics. In section 4 we explain the developed environment, the chromosome structure and the proposed fitness. Section 5 describes the evaluation of all performed experiments. We finish presenting the conclusion and the future perspectives.

MOBILE ROBOTICS

Several current works demonstrate mobile robotic usage (individual system) on hostile operations as the rescue auxiliary robot Raposa [9] and SACI robot [10] developed for acting on fire combat. Moreover, there are robots to perform tasks on aquatic environments, space, caves and volcanoes exploration, and even to household use. Multi-robotic systems should be formed by robots able to effectively act on tasks, so knowledge about robotic control is a very important field. Works describing intelligent robot control and navigation can be seen in [11] and [12]. In 2004 and 2005, DARPA Grand Challenge [13], financed and organized by the Defense Advanced Research Projects Agency, was a competition where the goal was building a completely autonomous vehicle that could complete a long way on dirt road on limited time. In 2007 the focus of the competition has changed, renamed to DARPA Urban Challenge, it had a new goal, to build a vehicle that could be autonomous on urban traffic, and execute tasks like parking, overtaking and intersection negotiations. These examples show trends in cooperation and multiple interactions.

The work with robotic groups adds a great number of possibilities on tasking-solving but brings a series of new questions to be solved in organization, coordination, collaboration and cooperation. Works using multi-robotic systems like [14] and [15] use pre-programmed rules on agents to perform formation. In [5, 16] are explored techniques to perform works with collectives robotics, used mainly for the purpose of applying the concept of self-organization and collective optimization, but task division is not directed explored. These work described in this section shows that the application of mobile robotics in control of incidents is an important and active topic of research and development. These several competitions also show that there is not yet a definitive or more adequate solution to the problem, and it is an open research field. In this research field there is no established consensus about conformation and actuation. Unpredicted situations with large degree of autonomy and robustness are still difficult to handle.

GENETIC ALGORITHMS

Genetic Algorithms (GA) [6], [7] are global optimization techniques that employ parallel and structured search strategies directed by fitness points search allowing multi-criteria search in a multi-dimensional space [17]. They are methods classified as unsupervised, which do not need any previous information database. The GAs use iterative procedures that mimic the evolution process of a population constituted by candidate solutions of a certain problem. The evolution process is random; however it is guided by a selection mechanism based on fitness of individual structures. For each algorithm iteration (single generation), a new set of structures is created by information exchange (bits or blocks) between selected structures of the previous generation [18]. The result of this conducts to the increasing in the individual fitness to the environment (better fitness = better problem solutions). A GA is structured in a way that the information about a determined system can be coded similarly to a biological chromosome, represented by a value array, where usually each sequence fragment represents a variable (gene).

When dealing with applications, [19] uses a GA to satisfactorily optimize trajectory planning for a robot arm. In [20], a GA model correctly evolves values for duration and forces application to actuators that allows a robot to walk. The work [21] presents a GA model to evolve the exploration method of a mobile robot in an unknown environment. These works present acceptable results for static environments; on the other side [22] describes a possible solution for operation in dynamic places, where the robot perform the navigation using GA. This robot is equipped with obstacle sensors and when identifies a possible collision, it stops and executes again the planning module using GA. In this way, the system becomes suitable for dynamic environments. In a previous work from the

author of this paper [23], it was proposed a GA for operation and control of robots in a dynamical simulation environment (fire propagation and combat), although in this previous work the author did not dealt with robot failures and fault tolerance.

GROUP FORMATION

4.1 Simulator

In order to build a real physical implementation of robotic systems, it is highly recommended to test the algorithms on virtual realistic simulation environments. Robotic system's simulation is especially necessary when dealing with big, expensive or fragile robots because it is a powerful tool to avoid wasting resources [3]. In our case, the proposed simulator should be able to reproduce an environmental disaster for a multi-robotic system performance. We propose the situation of a forest fire, so, in this case an intelligent vehicle squad (composed by a group of road graders) has the purpose of combat the forest fire acting by creating firebreaks around the fire.

The developed 3D simulation environment uses the OSG library [24] which is responsible by graphic output, the Demeter library [25] that is responsible by irregular terrain generation and ODE library [26] which is responsible by physics simulation realism, including the robot control and the collision detection/reaction involving the objects presents in the environment (e.g. robots, trees, terrain inclination). Using ODE library allow the physically simulated robots to comply with gravity, inertia and friction. Also, a 2D simulation environment was built to allow faster simulations (ignoring physical restrictions on robot navigation). This 2D prototype is implemented with SDL [27], as show the Figures 1 and 2. Both prototypes have the same fire propagation behaviour and use C/C++ as programming language.

The control variables of vegetation simulation and fire propagation are updated in a matrix representing terrain local properties. This matrix has for each terrain region the type of present vegetation; consequently, associating this information with the wind orientation and intensity we can build the fire propagation simulation. Regarding the wind, both its intensity and its orientation can be generated randomly or configured with parameters defined by the user. The time of permanence of the fire in an area is related directly to the present vegetation type and behaves in basis of terrain type values, terrain slope, wind orientation and intensity. In this way the fire spreading simulation tries to model the fire propagation as realistic as possible. To implement the fire spreading, we obtained from [28] real velocity measurements. The detailed characteristics about fire spreading modelled to this work, as well as the forest fuel models and the real operation techniques are compiled into [23]. The simulated terrain is based on topographical maps and on forest fuel maps models that can also be seen in [23].

The 3D simulation tool uses a realistic autonomous vehicle model (mobile robot), defining a 4 wheel vehicle (Fig. 4(b)) with steering and acceleration/break control (physically simulated: vehicle kinematics and dynamics controlled by ODE tool). The robots were configured with laser rangefinder simulated sensor used in order to detect and avoid obstacles. The vehicle is controlled by an Artificial Neural Network that reads the sensors (laser), the estimated position/orientation (GPS, compass) and generates the commands to the actuators (steering/acceleration). More details about vehicle development can be seen in [29].

Table 1: Chromosome structure (group of four robots – angle and radius related to the fire's starting point – using 2D polar coordinates relative to this point).

Gene	Function	Min. value	Max. value
0	Initial angle of robot 0	0.0°	360.0°
1	Final angle of robot 0; initial of robot 1	0.0°	360.0°
2	Final angle of robot 1; initial of robot 2	0.0°	360.0°
3	Final angle of robot 2; initial of robot 3	0.0°	360.0°
4	Final angle of robot 3	0.0°	360.0°
5	Initial radius of robot 0	10.0m	100.0m
6	Final radius of robot 0; initial of robot 1	10.0m	100.0m
7	Final radius of robot 1; initial of robot 2	10.0m	100.0m
8	Final radius of robot 2; initial of robot 3	10.0m	100.0m
9	Final radius of robot 3	10.0m	100.0m

4.2 GA Description

The GA optimizes fire combat position for each robot of the group, specifically: (i) Initial combat position for each member of the group (beginning point of firebreak creation); (ii) Final combat position for each member of the group (final point of firebreak creation). These positions are sent by command messages, used to activate the robots. To perform the simulations is necessary: (i) To know the available number of robots; (ii) To know robot's operation speed; (iii) To know robot's initial position; (iv) To have the ability to simulate fire propagation. To simulate fire

propagation is necessary: (i) To get initial fire position; (ii) To get wind direction; (iii) To get a simplified copy of the map (land and vegetation). This set of required information can be obtained by sensors. The proposed chromosome's structure has information about the entire group of robots. Therefore, we need 10 genes to a group of four robots. In the proposed structure of the chromosome, the final position of a robot is the starting position of the next, as show Table 1. The chromosome uses float point genome instead of binary genome due to the comparative analysis showed in [30] involving binary and float point representations; where it was presented that floating point representations have significant advantages, mainly related to precision and convergence speed.

$$x_d = x_a + r_i . \cos(a_i) \tag{1}$$
$$y_d = y_a + r_i . \sin(a_i) \tag{2}$$

The coordinates of operation are estimated applying Equations 1 and 2 to the best individual obtained by the GA. In these equations (x_d, y_d) is the robot's destination position, (x_a, y_a) is the starting position of the fire, r_i is the radius (gene 5 to 9) and a_i is the angle (gene 0 to 4). The radius and the angle are specific to each operation of each robot (initial and final coordinate of firebreaks creation).

4.3 Fitness
The fitness function guides the GA optimization. In this work, the proposed fitness is related with saved vegetation area and with combat units usage rate (coverage area per robot); therefore, the fitness accumulates: (i) Total burned area: trying to minimize burned area, (ii) Firebreak total area: trying to minimize robot's work area, avoiding to create firebreak on non-risk areas, (iii) Trying to minimize the difference among general average of useful firebreaks in relation to each individual useful firebreak, equalizing worked areas. The GA tries to minimize the fitness function value, which means less burned vegetation, less created firebreaks, and less difference between the sizes of firebreaks of each robot.

EXPERIMENTS AND RESULTS

5.1 Evaluation of GA Parameters
Considering that convergence speed is one of the most important aspects for the proposed system, several evaluations have been done to verify and find the best parameter set for the proposed GA. The parameter set evaluated on the algorithm is presented on Table 2. In order to verify the best parameter set, we set as fixed the climatic characteristics of the fire simulation and the initial robots positions to make the evaluations. Ten simulations were executed for each GA parameter set, which implies a total of 160 simulations. We made visual observations in 20% of the simulation results to verify which should be the ideal fitness for the proposed fire propagation/combat. The Figure 1 presents some simulation's visual results.

Table 2: Parameter set evaluated using the GA.

Parameter set	Selection scheme	Type of crossover	Type of mutation	Mutation rate
A	Tournament	Two-point	Uniform	10.00%
B	Tournament	Two-point	Uniform	50.00%
C	Tournament	Two-point	Gaussian	10.00%
D	Tournament	Two-point	Gaussian	50.00%
E	Tournament	One-point	Uniform	10.00%
F	Tournament	One-point	Uniform	50.00%
G	Tournament	One-point	Gaussian	10.00%
H	Tournament	One-point	Gaussian	50.00%
I	Stochastic	Two-point	Uniform	10.00%
J	Stochastic	Two-point	Uniform	50.00%
K	Stochastic	Two-point	Gaussian	10.00%
L	Stochastic	Two-point	Gaussian	50.00%
M	Stochastic	One-point	Uniform	10.00%
N	Stochastic	One-point	Uniform	50.00%
O	Stochastic	One-point	Gaussian	10.00%
P	Stochastic	One-point	Gaussian	50.00%

Figures 1(a) and 1(b) presents simulations sequences with fitness of 3,800 units; the fire was stopped but the firebreak is poorly optimized. Figure 1(c) presents a fitness of 3,480 units and Figure 1(d) presents a fitness of 2,643 units. The last two figures presents visual efficient operations but in mathematical terms the resulting chromosome analysis shows that in Figure 1(c) the standard deviation over actuation areas average is of 6.05 degrees and 2.37 radius units and Figure 1(d) standard deviation over actuation areas average of 3.38 degrees and 0.23 radius units. Both teams extinguished the fire but Figure 1(c) shows more equalized work areas among robots. Thus, from visual observations, fitness under 3,500 units was defined as acceptable and fitness below 3,000 units defined as ideal for this experiment.

From initial evaluations (Table 2) the parameter set that shows fitness below 3,000 units in, at least, 30% of the simulation were only sets {D, E, I, K, M}. None configuration presented more than 50% of fitness solutions below 3,500 units. Thus, a new set of simulations was performed, considering the five parameter's set that showed the best results but with an increase in the number of individuals (from 100 to 150) and in the number of generations (from 500 to 700).

(a) (b) (c) (d)
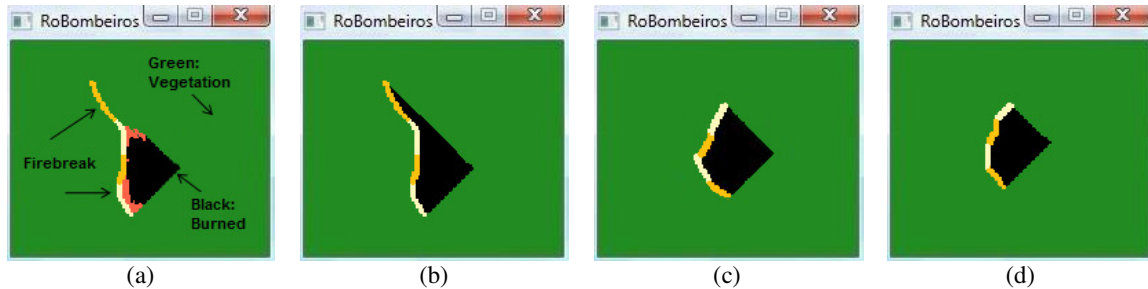
Figure 1: (a) and (b) Simulations sequences with fitness of 3,800 units; the fire was stopped but the firebreak is poorly optimized. (c) Simulation with fitness of 3,480 units. (d) Simulation with fitness of 2643 units.

Table 3 presents the results of second set of experiments. We can see that only the experience *I'* could obtain 100% of the fitness below 3,500 units. Therefore, the evaluations with reorganization will use the parameters of the experiment *I'* (two-point crossover, uniform mutation, mutation rate of 10% and stochastic remainder sampling selector).

**Table 3:** Results of the second set of evaluations.

Parameter set	Fitness < 3.000	Fitness < 3.500	Average	Standard deviation
D'	50%	80%	3,103.0	514.1
E'	70%	80%	3,139.6	699.4
I'	60%	100%	3,036.7	249.1
K'	40%	40%	3,764.7	1,049.8
M'	50%	90%	2,961.6	413.9

The two selection schemes evaluated in this work were Stochastic Remainder Sampling Selector and the Tournament Selector. In the Stochastic Remainder Sampling Selector, the selection is done in two steps: First, a new population is filled based on the value obtained by dividing the fitness of each individual by average fitness of the population. Values greater than 1 are inserted in the population and the remaining positions are filled based on probabilities considering the fractional part of each individual, then, the individuals are selected through random selection [30]. In the Tournament Selector, the selection in done using the Roulette Wheel Selection to obtain n individuals, then, pick the best one to put in the new population. In the GAlib [31], the Tournament Selector considers two individuals on each evaluation. Related to mutation, Gaussian and Uniform mutation were evaluated. In the first case, when a reasonable chromosome is identified, small adjustments are done (genes are changed by Gaussian distribution). In the second case, it changes a number to another random one, providing abrupt changes on values.

5.2 Reorganization

The robots are equipped with capabilities of sending and receiving messages. In the tactical operation simulation, we set a prefixed delay for exchanging messages between the robots and the coordinating agent (who was responsible for the evolution). The robots must send messages with a certain delay, so the coordinator agent knows about possible failure in the operation. Therefore, when identifying a problem, the coordinator agent may act in two ways:

(i) reorganize the whole group ignoring one of the members or (ii) send new robot units. We performed simulations in the two ways.


(a)   (b)   (c)   (d)
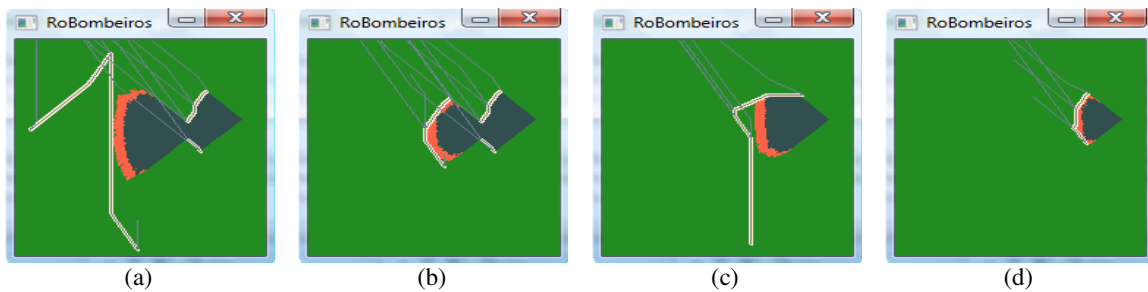
Figure 2: (a) and (b) Reorganization which sends two new robots. (a) Best individual on first generation, without heuristic. (b) Best individual on first generation, with the proposed heuristic. (c) and (d) Reorganization from four to three robots. (c) Best individual on first generation, without heuristic. (d) Best individual on first generation, with the proposed heuristic.

The reorganization might be done in the same way as was done the first organization, but considering possible changes in the environment caused by the robots. Due to time for GA evolution being long, we propose a heuristic in order to make the reorganization faster. In this heuristic, we propose a new set of individuals based on the best individual obtained by the first evolution. Using specific rules is not feasible due to the fact that the spread of fire is very sensitive to climatic conditions such as wind speed and direction, type of vegetation and terrain slope.

In the proposed heuristic for reorganization, an initial population is created based on the best individual obtained by the first evolution. In the reorganization from 4 to 3 robots (Figure 2(d)), after obtained the initial and final points of the original firebreak, we calculate the new positions considering the division of work among the three robots. Also, there are others individuals inserted in the GA population considering increase in angle and radius of operation, this increase is done in 2%, 4%, 8%, 12%, 16%, 24%, 32% and 40%, totalling 81 individuals that are used as initial population.

For the reorganization using two new robots, we use a similar idea presented in previous paragraph, but with some variation: when there is a failure in only one fragment of the firebreak (Figure 2(b)), the new individuals are generated based on the initial and final points of the operation of the robot that failed (not the full firebreak, as in the reorganization from 4 to 3). Also, the starting point of fire is considered to be the midpoint of where the crash occurred (not the original point). The Figure 2 presents reorganizations scenes. We can see that, even in the first generation, the proposed heuristic is able to obtain a formation to create an efficient firebreak. The evolution, even using heuristics, is important due to the fact of allowing a firebreak to be finely tuned with vegetation and terrain.
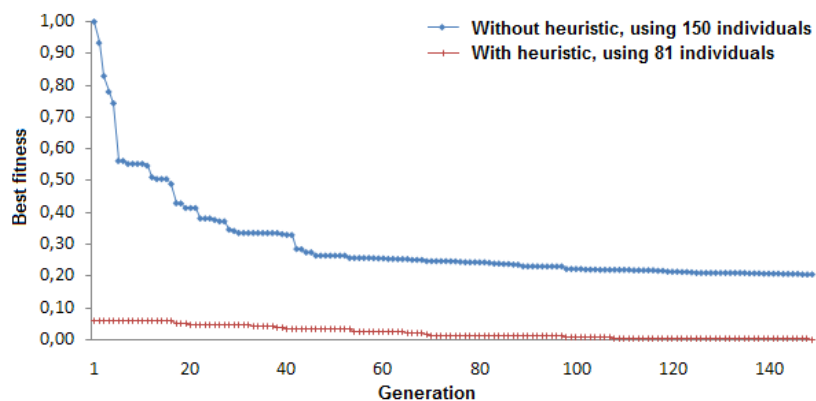
Figure 3: Evolution of the fitness according to number of generations when doing reorganization from four to three robots. The graphic shows fitness with and without the proposed heuristic.

Figure 3 presents fitness evolution graphic of the reorganization from four to three robots, showing the average of 10 simulations (using the parameter set of experience *I'*). Fire spread simulation considered East-West wind direction and relative wind speed at 7km/h; robot navigation speed of 35km/h; robots positioned on 2km far from fire starting point (fire propagation origin). The graphic shows that the use of the proposed heuristics allows to get, even in the initial population, an individual able to perform the fire fight. The Figure 4 presents satisfactory evolution result applied on 3D virtual simulation environment. The 3D prototype showed that robots completely surround the fire and create the firebreaks on a satisfactory way. The graphic of reorganization using two new robots

presented similar curves. Some simulations using different navigation speed and fire propagation were performed but on a small number of experiments. These experiments are not detailed on this text but also presented satisfactory results.

(a)   (b)
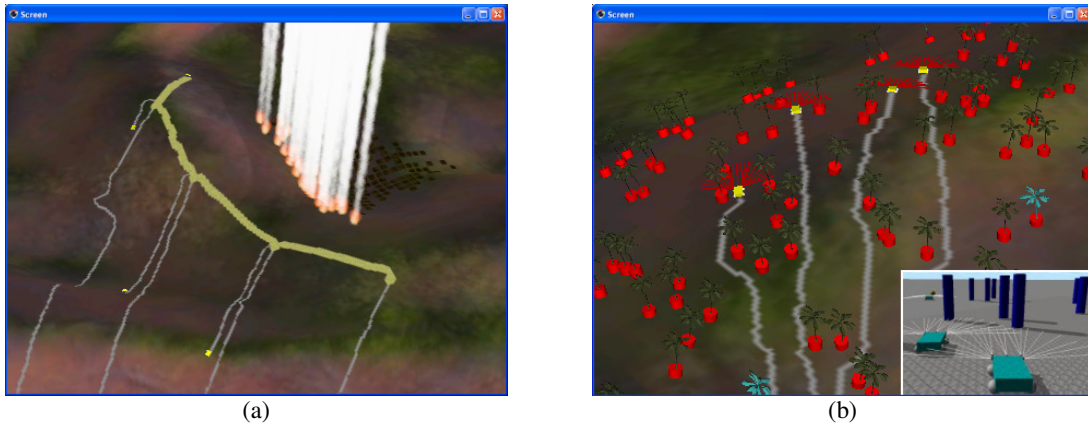
Figure 1: (a) Satisfactory results of the GA: four robots creating a firebreak. (b) Detailed view of navigation with obstacles avoidance.

FUTURE WORK

Some approaches are planned as future work: (i) a detailed study on others methods for robot coordination, such as Swarm-Based Models [5], [16] and Market-Based Approaches [32]; (ii) comparison of the efficiency of GA with Particle Swarm Optimization [33] and Simulated Annealing [34]. After these evaluations by simulation, real robots should be used in the experiments.

CONCLUSION

In this paper we described the evaluation of a fault-tolerant model for tactic operations of mobile robotic groups. The coordinated action of the group was planned with Genetic Algorithms and aims to act on an environmental disaster scenario, simulated as a forest fire. The robotic squad mission should surround the fire and avoid fire's propagation. Initially, we evaluated several parameters in the GA, seeking to obtain the set of parameters that would accomplish the more efficient evolution. Then, we simulated failures in robots operations in order to evaluate strategies of reorganization. The simulation's results showed that with an adequate set of parameters is possible to get satisfactory strategic positions to coordinate and to reorganize the robotic group in case of robot failures.

References

[1] Dudek, G., Jenkin, M. *Computational Principles of Mobile Robotics*. The MIT Press, 2000.

[2] Bekey, G.A. *Autonomous Robots: From Biological Inspiration to Implementation and Control*. The MIT Press, 2005.

[3] Go, J., Browning, B., Veloso, M. Accurate and exible simulation for dynamic, vision-centric robots. In: *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems* (AAMAS), IEEE Computer Society. p. 1388-1389. 2004.

[4] Osagie, P. *Distributed control for networked autonomous vehicles*. Master's thesis, KTH CSC, Royal Institute of Technology, Sweden, 2006.

[5] Mondada, F., Gambardella, L.M., Floreano, D., Dorigo, M. The cooperation of swarm-bots: Physical interactions in collective robotics. *IEEE Robotics and Automation Magazine*, 12:21-28, 2005.

[6] Holland, J. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, 1975.

[7] Mitchell, M. *An Introduction to Genetic Algorithms*. The MIT Press, 1996.

[8] Pessin, G., Osório, F.,Wolf, D.F., Dias, M.A. Genetic algorithm applied to robotic squad coordination. In: *Proceedings of Electronics, Robotics and Automotive Mechanics Conference* (CERMA), IEEE Computer

Society, 169-174, 2009.

[9] IdMind: Projecto raposa, http://raposa.idmind.pt, 2007.

[10] Macedo, A.R.M., Macedo, A.R.L., Duarte, J.B.F.: Robótica aplicada ao combate a incidentes. *Revista TN Petróleo* 53:108-113, 2007.

[11] Zhao, Y., Collins, E.G. Robust automatic parallel parking in tight spaces via fuzzy logic. *Robotics and Autonomous Systems* 51:111-127, 2005.

[12] Heinen, M., Osório, F., Heinen, F., Kelber, C. Seva3d: Using artificial neural networks to autonomous vehicle parking control. In: *International Joint Conference on Neural Networks* (IJCNN), p. 4704-4711, 2006.

[13] Darpa: Grand challenge webpage, www.darpa.mil/grandchallenge, 2007.

[14] Yamaguchi, H. Adaptive formation control for distributed autonomous mobile robot groups. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), p. 2300-2305, 1997.

[15] Balch, T., Arkin, R.C. Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation* 14:926-939, 1998.

[16] Dorigo, M., Trianni, V., Sahin, E., Gro, R., Labella, T., Baldassarre, G., Nol_, S., Deneubourg, J., Mondada, F., Floreano, D., Gambardella, L. Evolving selforganizing behaviors for a swarm-bot. *Autonomous Robots* 17:223-245, 2004.

[17] Carvalho, A.C.F., Braga, A.P., Ludermir, T.B. *Computação Evolutiva*. In: Sistemas Inteligentes: Fundamentos e Aplicações, p. 225-248, 2003.

[18] Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Professional, 1989.

[19] Nguyen, V.B., Morris, A.S. Genetic algorithm tuned fuzzy logic controller for a robot arm with two-link exibility and two-joint elasticity. *Journal of Intelligent and Robotic Systems* 49(1)3-18, 2007.

[20] Heinen, M.R., Osório, F.S. Applying genetic algorithms to control gait of simulated robots, *Proc. of Fourth Congress of Electronics, Robotics and Automotive Mechanics*, IEEE Computer Society, 2007.

[21] Zhou, Y. *An area exploration strategy evolved by genetic algorithm*. Master's thesis, Univ. of Georgia, 2005.

[22] Burchardt, H., Salomon, R. Implementation of path planning using g.a. on mobile robots. In: *IEEE Congress on Evolutionary Computation*, 2006.

[23] Pessin, G.: *Evolução de estratégias e controle inteligente em sistemas multirrobóticos robustos*. Master's thesis, Universidade do Vale do Rio dos Sinos, 2008.

[24] OSG. Open scene graph community, http://www.openscenegraph.com, 2009.

[25] Demeter. Demeter terrain engine, demeter.sourceforge.net, 2009.

[26] Smith, R. Open dynamics engine website, www.ode.org, 2009.

[27] Sdl. Simple directmedia layer website, www.libsdl.org, 2009.

[28] Koproski, L.P. O fogo e seus efeitos sobre a herpeto e a mastofauna terrestre e a mastofauna terrestre no parque nacional de ilha grande. Master's thesis, UFPR, 2005.

[29] Pessin, G., Osório, F., Hata, A.Y., Wolf, D.F. Intelligent control and evolutionary strategies applied to multirobotic systems. In: *Proceedings of IEEE International Conference on Industrial Technology* (ICIT), IEEE Computer Society, 2010.

[30] Michalewicz, Z. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, 1996.

[31] Wall, M.: Genetic algorithms library, http://lancet.mit.edu/ga, 2009.

[32] Dias, M.B., Zlot, R., Kalra, N., Stentz, A. Market-based multirobot coordination: A survey and analysis. *Proceedings of the IEEE* 94:1257-1270, 2006.

[33] Kennedy, J., Eberhart, R. Particle swarm optimization. In: *Proceedings of IEEE International Conference on Neural Networks*, IEEE Press, 1995.

[34] Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P. Optimization by simulated annealing. *Science* 220:671-680, 1983.