Adaptively Coordinating Heterogeneous Robot Teams through Asynchronous Situated Coevolution

Abraham Prieto, Francisco Bellas, and Richard J. Duro

Integrated Group for Engineering Research
Universidade da Coruña, Spain
{abprieto,fran,richard}@udc.es

Abstract. Adapting to changing situations and objectives and selforganazing without a central controller in order to achieve an objective has become one of the main challenges in the design and operation of multirobot systems. The Asynchronous Situated Coevolution (ASiCO) algorithm has been successfully applied in surveillance tasks defined by just one global objective. In this paper we present the results obtained with ASiCO in more complex multirobot problems with several objectives that require a heterogeneous population of robot controllers that autonomously distribute the tasks. The paper focuses on the benefits of evolving an affinity coefficient that characterizes the individual genotypes.

Keywords: Coevolution, Adaptation, Multi-robot Systems, Coordination.

1 Introduction

This work is concerned with the application of a real time coevolutionary strategy, based on Watson et al's [1] Embodied Evolution (EE) concept that provides a means for groups of robots to selforganize and perform tasks in an efficient manner. The inspiration for this approach comes from the field of Artificial Life, but we have included some of the notions of utility functions found in the multiagent systems literature. In the original implementation of EE, the authors sought to establish a completely distributed evolutionary algorithm embodied in physical robots. Their approach was based on the hypothesis that a large number of robots could be used for the evaluation stage of an evolutionary process devoted to obtaining a controller for a particular task. This led to a set of ideas and design requirements that had to be taken into account, such as the fact that the evolutionary process had to be decentralized and thus the evaluations required for the determination of the fitness of an individual should take place directly within the individual in an embodied and localized manner, preferably on the physical robot itself. It is clear that this differs radically from other strategies found in the Evolutionary Robotics (ER) literature [2][3][4] where a centralized evolutionary algorithm is in charge of carrying out whatever operations are necessary using information from all the robots in a simulation (or even in some cases in real robots), and usually off-line, in order to obtain a controller that when instantiated on the robots produces the desired individual or group behavior.

Lately, a lot of effort has been devoted to the generation of coordinated behaviors for large (and sometimes not so large) groups of robots. Some authors have been

C.S. Leung, M. Lee, and J.H. Chan (Eds.): ICONIP 2009, Part II, LNCS 5864, pp. 75–82, 2009.

working on the formalization of the problems [5] in order to produce hand crafted algorithms or controllers for particular tasks while others have concentrated on implementation issues [6][7][8]. However, many of these approaches are particular to a task (i.e. foraging or flocking) or environment, often using homogeneous sets of robots and/or controllers and do not provide a general framework for obtaining collective behaviors. Some work is beginning to appear where some problems are characterized and a study of the production of collective solutions is carried out in order to determine the most appropriate for each case. For an example see [9]. However, it is often ignored that the structure of most real world problems does not allow for an easy decomposition into subproblems , nor are they known beforehand and, consequently, the robot teams that solve them must arise in a joint manner. This is especially so in the case of complex dynamic problems where the environment or even the objectives change with time introducing a new level of complexity and a requirement for real time autonomous adaptation on the part of the robot teams.

The objective of this paper is to present the application of Asynchronous Situated Coevolution (ASiCo) as a valid distributed and adaptive strategy in order to allow for groups of robots to selforganize to perform tasks in an efficient manner. ASiCo draws inspiration from the main features of some Artificial Life based distributed evolutionary approaches but includes some ideas from the multiagent systems literature to provide a way to implement the objectives of the collective system through the creation of energy and interaction based utility distribution schemes. A first application to simple in multirobot surveillance tasks may be found in [10].

The following sections are devoted to providing a brief description of the algorithm as well as some important operators, such as embryonic based reproduction, required for the adaptation of the algorithm to real time operation on distributed multirobot systems. After this, we will show some results obtained from the application of this approach to the problem of obtaining controllers for a set of robots that must selforganize to perform a collective cleaning task. The objective is then made more difficult by having two different tasks that the robot team must concurrently perform, thus leading to the speciation of heterogeneous controller populations. This necessary speciation is made more efficient through the inclusion of an affinity term in the reproduction mechanism. Finally, some conclusions are presented.

2 A Brief Description of Asynchronous Situated Coevolution

The ASiCo algorithm is inspired on Artificial Life simulations in terms of the use of decentralized and asynchronous open-ended evolution. Unlike other bio-inspired approaches such as genetic algorithms in which selection and evaluation of the individuals is carried out in a centralized manner at regular processing intervals based on an objective function, this type of evolution is situated. This means that all of the interactions in the population are local and depend on spatial and temporal coincidence of the individuals in the simulation environment, implying an intrinsic decentralization. Consequently, reproduction, creation of new individuals or their elimination, is driven by events that may occur in the environment in a decentralized way.

This type of evolution has usually been employed for analysis purposes, this is, to study how a system evolves in an open-ended manner, and not really with an engineering objective in mind, and thus, there is no clear procedure to relate the global objective to be achieved with the local objectives of the agents that participate in the process. To this end, inspiration is taken from the studies of utility functions and their distribution among individuals in order to structure the energy dynamics of the environment to guide evolution to the objectives sought. Specifically, we have used the principled evaluation function selection procedure for evolving coordinated multirobot systems developed by Agogino and Tumer [11], which establishes a formal procedure to obtain the individual utility function from the global function. With this procedure, ASiCo open-ended evolution becomes an evolutionary optimization algorithm that provides a distributed solution by means of the whole population and not only by the best individual as in typical evolutionary algorithms.

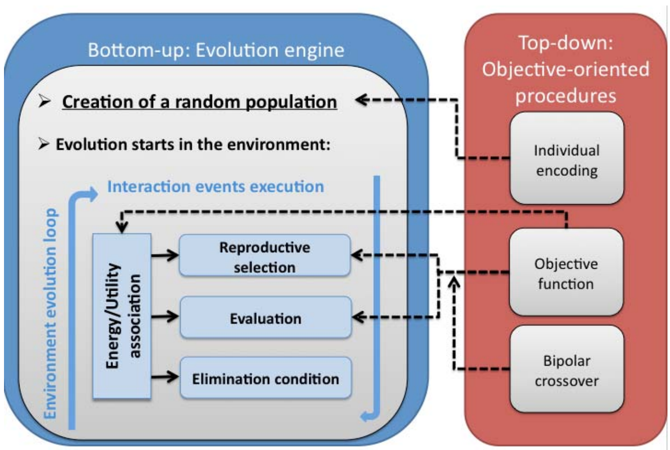

Fig. 1. Schematic representation of ASiCo structure

Fig. 1 displays a schematic representation of the algorithm's structure divided into two different parts, and the relations between these procedures and the processes carried out during evolution. On one hand (right block) we have the procedures that guide the evolution towards an objective. *Individual encoding* defines the solutions that can be generated, the *objective function* is established using the utility functions commented above, and, finally, *bipolar crossover* allows for the evolution of a heterogeneous population. On the other hand (left block) we have the *evolution engine*, which is based on the interactions among elements in the environment. After the *creation of a random population*, the execution of the interaction events occurs in a continuous loop modifying the state of the elements. In some conditions and based on energetic criteria for spatiotemporally coinciding individuals, the procedures that represent the evolution of the population (*selection*, *evaluation* and *elimination*) occur. The *energy/utility association* represents the energetic rules in the environment and affects these three procedures. Finally, the objective function defines the energetic criteria, the selection and the elimination.

Thus, ASiCo is an interaction driven algorithm. Interactions are a set of rules that make the state of the elements and individuals in the environment change in time due to particular events. This process is independent from the evolution of the population. Two elements are very relevant within ASiCo. On one hand the energy flow, that is, we have different rules that use the concept of energy to manage, through a bio-inspired strategy, the assignment of quality and the conditions to the creation and deletion of individuals in the population. On the other, reproductive selection is the set of rules that regulates the reproduction process. This selection process must be defined for each problem and is based on spatial interactions together with some energetic criteria. Specifically, it is usually performed by means of a tournament operator, which, in a typical evolutionary algorithm, randomly selects a number of individuals from the population for the reproduction. This centralized behavior is not possible here, so the tournament has been modified to be *asynchronous and decentralized* and, consequently, based on local interactions between the individuals.

A very important aspect of ASiCo is reproduction. This mechanism needs to be adapted to the objectives sought and in this paper we are interested in groups of real robots with a fixed number of individuals. Consequently, as we do not want real robots to "die" nor can we make robots "appear" in the environment, the reproduction mechanism has to contemplate the fact that the number of robots is fixed and still provide a way for the population to evolve in a distributed manner. Some authors have proposed different strategies, such as PGTA [1], which imply continuously modifying the genetic make up of the robots on line. This leads to instabilities in robot behaviors due to lack of evaluation time in the environment. Another approach, which is the one followed here, is to synchronize death with birth. In fact, as the robots are preset, we can only work with their controllers and thus, a death-birth process within a robot is just a change of its controller. To allow for evolutionary pressure within this process, we have designed a reproduction mechanism for fixed size populations called Embryo Based Reproduction (EBR).

The idea behind EBR is that each agent, carries, in addition to its own parameters, another set of parameters corresponding to its embryo and an associated pre-utility value for the embryo that estimates the utility of the agent generated from it. Thus, when a new agent is created, its embryo is generated as a mutation of the parent genotype with half of its energy. During the life of an agent, the embryo is modified whenever the agent meets another one and evaluates it positively, meaning that the average of the utility of the two parents is higher than the pre-utility of the current embryo. Finally, when the parent dies because it ran out of energy or time or for whatever other reason, the embryo substitutes the parent, that is, the control of the robotic unit is assumed by the embryo and a new embryo is generated within the robot. This way, we ensure that the size of the population remains constant and that the process takes place in an asynchronous and decentralized manner.

3 Experiments and Results

This section contains some of the results of applying ASiCo to the problem of obtaining controllers for a set of 20 preset robots that must carry out a task cooperatively. In the particular example we consider, the task has to do with cleaning an arbitrary area. Other authors have studied this problem using different algorithms [12][13], but our experience in multirobot surveillance tasks where ASiCO has provided better results as compared to typical evolutionary strategies [10], has led us to apply it in cleaning tasks too and thus analyze its capabilities in a typical testbed for multirobot systems.

The particular experiments we have developed consider the cleaning problem in a progressively more complex environment. It is made up of cells and each cell is characterized by a level of dirt that increases through a non-linear function while no robot cleans it. The dirt level goes to zero when the cell is cleaned and starts to increase again until it is re-cleaned by another robot. The final objective of the multirobot system is to keep the entire environment as clean as possible all the time. In terms of sensing, it is very primitive. A robot can detect how many robots are within the sensor reach (which is a constant for all the robots) in each of the four quadrants around it, the dirt level of the cell it is on and if a collision has occurred. In addition, each robot has a small memory that allows it to remember its position n instants before. In terms of actions, all of the robots move at the same speed and the control system just provides a value for the angular velocity (rate of turn).

The control system is based on a RBFNN (Radial Basis Function Neural Network) whose parameters are encoded in the genotype of the robot together with the length of the memory. The inputs to this network correspond to the number of robots around it within the four quadrants, the module and angle of the vector relating its position to that n instants before, the dirt level and its current position and the value of its collision sensor (whether a collision has occurred). The global utility is the sum of the dirt level in all the cells, and must be minimized. The individual utility is the sum of dirt levels in the cells in the instant the robot cleaned them.

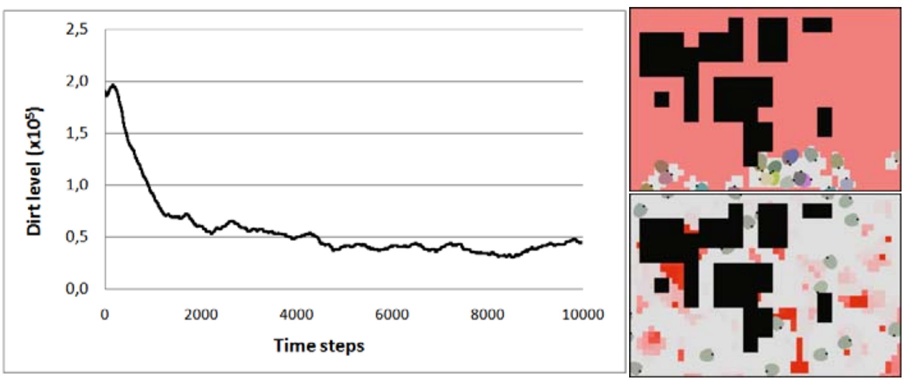

Fig. 2. Evolution of the global dirt level (left) and two screenshots of the simulation environment in simulation step 0 (top right) and simulation step 8000 (right bottom). Red intensity indicates dirt level.

Fig. 2 left shows the evolution of the global dirt level in an environment that is being controlled by a set of 20 robots as described above. As we can see, it decreases to a stable level in around 5000 simulation steps. To have a visual idea of the fulfillment of the task, the right images of Fig. 2 show two screenshots of the simulator. Different robot colors represent different genotypes and red intensity represents the dirt level. The top image shows the initial situation, where all the controllers are random. The bottom image corresponds to the state after around 8000 simulation steps. It is clear that the controllers of the robots have improved to the point of being able to obtain quite a low dirt level (an average dirt level of around 35000 units that, taking into account that the environment has 1584 cells, corresponds to 22 units per cell, consequently, every cell is being explored every 363 simulation steps). The fact that the whole area is monitored efficiently is a consequence of the emergence of a coordination strategy forced by the global utility requirements. Another interesting result that can be seen in Fig. 2 bottom right is that the robots tend towards a homogeneous genotype in this task (all the robots have the same color). This seems reasonable given the fact that the environment is relatively homogeneous in terms of requirements for the agents.

Once we have seen that ASiCO is able to solve the basic cleaning setup, we decided to study a more complex environment where the type of cleaning could be different depending on the zones. To simulate this situation we have designed an environment with two separate areas and provided the robots with a new sensor that detects if the area must be swept or vacuumed (we assume that all the robots can perform both tasks). The final result is shown in the screenshot of Fig. 3 left in simulation step 30000. As we can see, two different species have been created providing a very low global dirt level, represented in Fig. 3 right. This figure shows, in addition, the dirt level of the swept and vacuumed zones where we can appreciate initial fluctuations in the population towards robots that are capable of solving both tasks, and consequently, there is a sharp decrease in the dirt level in one zone or the other, but a high global dirt level. Around simulation step 6000, the population is divided into two different species and, consequently, both dirt levels decrease.

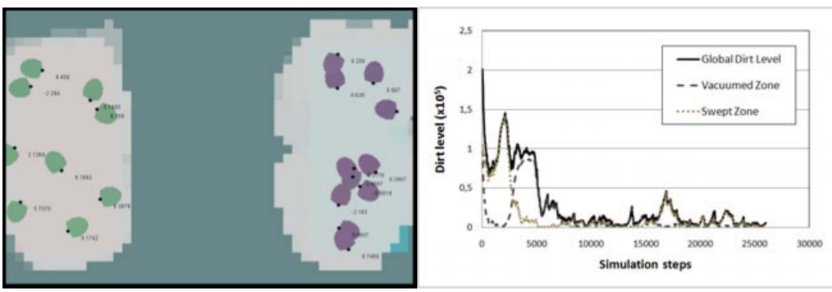

Fig. 3. Population of robots in simulation step 30000 when the cleaning problem is divided in two different tasks (left). Evolution of the global dirt level and that in both zones (right).

ASiCO automatically selforganizes the population into two different types of robots. This is a very interesting capability of the algorithm that can be used in a multi-robot system design stage to automatically obtain the distribution of robots into types, in this case, with 11 vacuum robots and 9 sweep robots. This result is a consequence of the Bipolar crossovers, which does not favor the appearance of homogeneous genotypes, and the spatial separation of the two zones, that leads to robots executing small movements that keep them within a small area. Obviously, in a general problem, this spatial separation may not exist but the two tasks would be present, and ASiCO should provide different species too.

In Fig. 4 we have represented the environment created to study the emergence of species without such spatial separation of the zones. In the top left image, we show the final result obtained when we place together the two zones that in Fig. 3 left were separate. As we can see, the solution provided by ASiCO is a single species, so it seems that the basic implementation of the algorithm needs a spatial separation to obtain heterogeneous genotypes. To solve it, we have included a new coefficient in the genetic selection process, called affinity. This coefficient is calculated using the average difference of all the genes of two individuals, that is, it represents the genotypic distance of two individuals. During evolution, this coefficient is used to promote the selection and crossover of genotypically similar individuals, thus favoring the creation of species. In the case that the task does not require speciation, this coefficient does not affect the evolution.

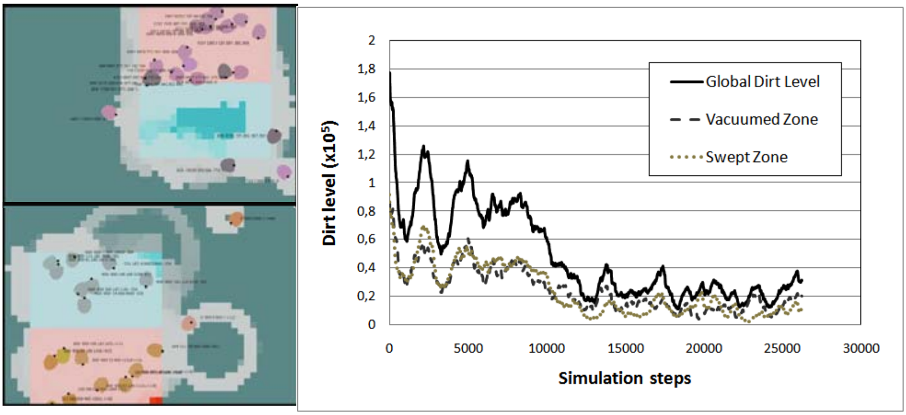

Fig. 4. Final multirobot system obtained when the two different areas that must be cleaned are placed together (left). The top image corresponds to the result obtained without affinity coefficient and the bottom one with it. Evolution of the dirt level (right).

After including the affinity coefficient, the result obtained is shown in Fig. 4 bottom left, where we can see two different species again. Fig. 4 right represents the evolution of the dirt level for the two zones and the global one. Now, the transient period that appeared in Fig. 3 right does not exist and all the dirt levels decrease from the beginning. With this improvement, the ASiCO algorithm is able to automatically provide a solution to the cleaning task with heterogeneous robots, specialized in sweeping or vacuuming.

4 Conclusions

The results presented in this paper provide an indication that Asynchronous Situated Coevolution (ASiCo) together with an embryonic like delayed reproduction mechanism (EBR) considering an affinity parameter is an effective approach for the introduction of real time evolution within robot coordination structures. The experiments here started from the premise that we had a fixed number of real robots and that the control strategy of the group had to arise in real time from their local interactions in a decentralized and asynchronous manner. It is clear from the results that, depending on the requirement of the problems, homogeneous or heterogeneous populations of controllers are obtained quite fast that allow the robot team to jointly achieve their objective and that when the objective or environment changes they adapt promptly to the new situations and objectives. The approach has been explored in a simulated environment that mimics the cleaning tasks that need to be carried out in real environments obtaining successful results.

Acknowledgments. This work was partially funded by the MEC of Spain through projects DEP2006-56158-C03-02 and DPI2006-15346-C03-01.

References

1. Watson, R.A., Ficici, S.G., Pollack, J.B.: Embodied Evolution: Distributing an Evolutionary Algorithm in a Population of Robots. Robotics and Autonomous Systems 39(1), 1–18 (2002)
2. Harvey, I.: Artificial Evolution and Real Robots. In: Sugisaka, M. (ed.) Proceedings of the International Symposium on Artificial Life and Robotics (AROB), Beppu, Japan, pp. 138–141 (1996)
3. Mataric, M.J., Cliff, D.: Challenges in Evolving Controllers for Physical Robots. Journal of Robotics and Autonomous Systems 19(1), 67–83 (1996)
4. Floreano, D., Mondada, F.: Evolutionary Neurocontrollers for Autonomous Mobile Robots. Neural Networks 11(7-8), 1461–1478 (1998)
5. Kolling, A., Carpin, S.: Multi-robot Surveillance: an Improved Algorithm for the GRAPH-CLEAR Problem. In: Proc. 2008 IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, pp. 2360–2365 (2008)
6. Burgard, W., Moors, M., Stachniss, C., Schneider, F.E.: Coordinated multi-robot exploration Robotics. IEEE Transactions on Robotics 21(3), 376–386 (2005)
7. Fox, D., Ko, J., Konolige, K., Limketkai, B., Schulz, D., Stewart, B.: Distributed Multirobot Exploration and Mapping. Proc. of the IEEE 94(7), 1325–1339 (2006)
8. Folgado, E., Rincón, M., Álvarez, J.R., Mira, J.: A Multi-robot Surveillance System Simulated in Gazebo. In: Mira, J., Álvarez, J.R. (eds.) IWINAC 2007. LNCS, vol. 4528, pp. 202–211. Springer, Heidelberg (2007)
9. Waibel, M., Keller, L., Floreano, D.: Genetic Team Composition and Level of Selection in the Evolution of Multi- Agent Systems. IEEE Transactions on Evolutionary Computation (to appear, 2009)
10. Schut, M.C., Haasdijk, E., Prieto, A.: Is Situated Evolution an Alternative for Classical Evolution? In: Proceedings CEC 2009, pp. 2971–2976 (2009)
11. Agogino, A., Tumer, K.: Efficient evaluation functions for evolving coordination. Evolutionary Computation 16(2), 257–288 (2008)
12. Chaomin, L., Yang, S.X., Stacey, D.A.: Real-time path planning with deadlock avoidance of multiple cleaning robots. Robotics and Automation 3, 4080–4085 (2003)
13. Wagner, I.A., Altshuler, Y., Yanovski, V., Bruckstein, A.M.: Cooperative Cleaners: A Study in Ant Robotics. The International Journal of Robotics Research 27(1), 127–151 (2008)