Artificial Life Optimization over Complex Networks

Mauro Annunziato[1], **Ramon Huerta**[2], **Matteo Lucchetti**[3], **Lev S. Tsimring**[2]

[1]Casaccia Research Centre, ENEA, Via Anguillarese 301, 00060 Rome Italy,
mauro.annunziato@casaccia.enea.it
[2]Institute for Non Linear Science, University of California, San Diego, La Jolla, CA, 92093 USA
{rhuerta,ltsimring}@ucsd.edu
[3]Department of Computer and System Science, University of Rome La Sapienza, Via Eudossiana 18, 00184 Rome Italy
lucchetti@dis.uniroma1.it

Abstract

Different topologies for the life space of an artificial life environment facing continuous optimization problems are explored. Starting from a lattice we tested the efficiency of the algorithm when the physical space in which the agents evolve is represented first by a higher degree lattice, then by a small world network, next by a random graph and last by a scale-free network. Interesting results are obtained about efficiency improvement with respect to the lattice in all cases, but the most relevant are those ones connected to the scale-free topology. Even if obtained on a particular implementation of an evolutionary algorithm, results shown are supposed to hold for a larger class of artificial environment and/or evolutionary contexts implemented in literature. First empirical explanations of the observed phenomena are given.

Keywords – Artificial life, complex networks

1. Introduction

Much effort has been devoted during recent years to developing evolutionary algorithms (EAs) with a physical space for individuals [1].

Artificial evolution within a physical space has been studied mainly theoretically for hard optimization problems, with the goal of creating a spatial structure in the population of solutions so to allow the formation of certain topologies for different groups of individuals. Structuring the population in space has been empirically shown to often improve the numerical and runtime behavior of the algorithm. In the past there have been some attempts to relate the physical space in which the individuals evolve (see for example [3]) with the efficiency of the global optimum reaching.

Moreover it is worth mentioning that the introduction of space in an evolutionary algorithm helps to create the so-called evolutionary niches. Those can be regarded as isolated regions, which preserve some currently bad solution that could be useful in the future, for it could be the ancestor of the global optimum both in static and dynamic fitness landscape. This allows to avoid a premature convergence toward a local minimum, which is one possible drawback of the genetic algorithms approach, and also maintains diversity in the population, which is a relevant issue in case of dynamically changing fitness landscapes.

Introducing the physical space in evolutionary computation leads to the simulation of contexts which are similar to the real interactions between living agents. All those are usually studied in the field of artificial life.

Recently the artificial life environments have began to be used in real world applications, with the hope to exploit both their adaptivity to environmental changes and their ability to reach the optimum (see for example [8]). In particular, optimal control and management of complex systems, such as industrial manufacturing processes (for a review see [6]) or resources allocation seem to be a very promising field.

In this paper we want to analyze the effect of the topology of the physical space in which the agents live in the effectiveness of the optimization process. A first attempt to study a different space topology in artificial life contexts has been recently made in [7], who studied the evolution of the Game of Life on a small world network instead of the regular lattice. In this work, building upon the last achievements in the field of complex networks, we consider an artificial life environment, which is meant to be as general as possible and we let the individuals move in a random way on different kinds of networks. Individuals evolve on four types of networks: the regular lattice, the random graph, the small world and the scale-free network.

In particular we mainly focused our attention on this last, which outperformed all the others. These types of networks are receiving great attention in the physical community, because many networks have been reported recently to follow a scale free degree distribution. Just as examples we can cite the Internet, the WWW, the e-mail network, metabolic networks, trust network and many more[5]. Their inspiring philosophy could be synthesized in the sentence '*the rich gets richer*', because each node has a probability to get a new link that is proportional to the number of its current link. Our experimentation pointed out some interesting remarks about the improvement in efficiency of global optimum reaching, which could be considered of general relevance.

In the next paragraph we will briefly review the main aspects of the artificial life environment implemented and

we will present the different topologies used. In the following we will show and discuss the main simulation results and at last we will give a glance to possible future developments of this research.

2. Model and method

The Artificial Life environment we study in this paper is the one introduced in [2], with some minor modifications (fig.1). This environment has been successfully used in a good deal of real world online optimization applications involving non-stationary dynamical systems. As a consequence, it has not been refined to tackle static optimization problems like the ones we are going to face here and its overall performance doesn't reach the state of the art of this field. Nevertheless it is good for our aim, because we are interested in a general feature whose impact holds for several other types of artificial life and/or evolutionary optimization environments.

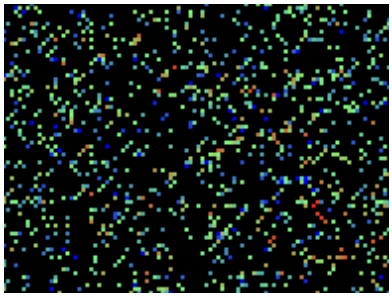

Figure 1 – The Artificial Life environment

The original artificial environment is a two-dimensional lattice, in which a randomly generated initial population of potential solutions to the problem is randomly distributed. In the following we will address with the term fitness the value of the cost function calculated for the current solution. During each iteration (a-life cycle), every individual moves around and can meet some other individual, reproduce, or die.

In our implementation each individual is characterized by its state, its genotype and the information about the solution it carries. The state takes into account all those characteristics that change during its life, such as for example the position in the 2D space. The genotype stores all the characteristics that do not change during the evolution, such as the mutation rate or the probability to reproduce. About the solution, it doesn't change during the individual life and is stored as a real vector in the genotype.

In order to manage the interactions between them we introduced a level of energy, which varies during the life of the individuals according to the following two simple rules. When there is a meeting between two individuals, the one with higher fitness gets a fixed amount of energy from the one with the lower. In such a way the low energy individuals are supposed to be the one corresponding to the worse current solutions. When there is a birth, a fixed amount of energy is given from the father to its son, so that only the individuals with energy higher then this threshold can reproduce, and this creates selection in the population. The son is added to the population and it is randomly located in a cell of the life space connected to the father's one. This is the only genetic operator we chose to implement and corresponds to a sort of $(\mu+\lambda)-ES$.

In their random motion in the space, the individuals are not permitted to collide in the same cell. During each cycle, a cell is chosen randomly among the neighbors of the current individual's location and if it is empty the individual is moved there; if in that location there is another individual, a meeting occurs, but neither the former nor the latter agent move.

The reproduction mechanism has been chosen as simple as possible, in order to get some general result not too connected with our specific implementation. It is haploid, so that with a fixed probability at each cycle just one individual is needed to generate another. The mutation mechanism is fixed during the evolution and does not adapt to the decreasing distance from the optimum. As a consequence, mutation amplitude is randomly chosen in an evenly distributed fixed interval around the solution that the father points to. Only one mutation per generation in the genotype of the parents is allowed. The newborn individuals are added to the previous population and do not replace their parents.

When an individual reaches null energy, this means that it lost several meeting, so it dies and is removed from the environment, freeing the cell where it was located in.

These simple rules, together with an opportune adjustment of the interaction parameters between the agents, have been empirically proven to generate equilibrium in the evolution, since the population reaches a dimension which fluctuates around a constant value, never going to saturation (life space full) or to extinction (empty space). This last issue allows the environment to always present a good diversity, i.e. several individuals different from the current optimal one, which is a characteristic that is useful in case of dynamic fitness landscapes.

As we said, we focus our attention on different topologies for the space in which the individuals evolve. We suppose that each agent performs a random walk in the space, and when considering different topologies, we just let the same rules explained above for birth, interaction, reproduction and death. As we will show in the next section, the topology of the space influences the meeting policy, and this suffices to produce different regimes in the environment and different efficiencies.

We start our analysis from the lattice, which is by far the most widely used structure for artificial environments implemented in literature, principally because of its similarity with the physical space. In a lattice each cell is linked with the surrounding cells up to a fixed scope. If we consider a physical 2D space of dimension $m \times n$, we can

map it on a network with $m \times n$ nodes, one for each different cell. In the case of the lattice, this network is regular, its degree k (that is the number of links for each node) is equal to the number of the cells surrounding the one we consider up to the fixed scope and its adjacency matrix is k-diagonal. For each individual, at each time step, we know the position (x, y) in the space, and so the node of the network and the line in the adjacency matrix which it points to. Hence we can compute its next location on the lattice by randomly choosing among the connected cells, i.e. among the ones on that line of the adjacency matrix. In particular, we considered two regular square lattices, with periodic boundary conditions, such that each site is linked to its 4 nearest neighbor for the first (up, down, left and right) and to its 8 nearest and corner-nearest neighbors for the second (fig. 2).

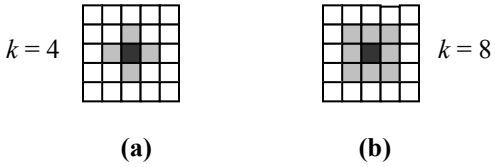

Figure 2 – Neighbors in the two considered lattices

For the second lattice structure (fig. 2(b)), with degree $k = 8$, following a common approach, we built a small world network. To do that, we rewired at random each link in the original lattice, with probability p, to another site of the network. Of course, with probability $(1 - p)$ the original link is preserved. Multiple connections are not allowed as well as self-connections. A small world topology corresponds to small p $(p = 0.001)$. For $p = 1$ we get a quasi-random graph. As we will show, when considering a random graph instead of a lattice, a higher performance of the algorithm is achieved. The small world topology produces intermediate results and we will not focus our attention on it.

Last, we built the space in the form of a network with a scale-free degree distribution, i.e. such that the fraction of sites having k connections follows a power law $P(k) = ck^{-\lambda}$ (for a review of scale- free networks see [4]).

To obtain a scale-free network we started with a small number of nodes (0.1% of the total), randomly chosen among the available ones and randomly connected among each other. We then took the remaining nodes sequentially and created 8 links for each of those (to have the same connectivity degree with respect to the previous three networks) in such a way that the probability for node m to be linked to node n is proportional to the degree of n (*preferential attachment*). With such a procedure we obtained $\lambda = 3$, as can be seen in fig. 3, where we plot the log-log graph of the degree distribution. This case brought a remarkable improvement in the algorithm efficiency.

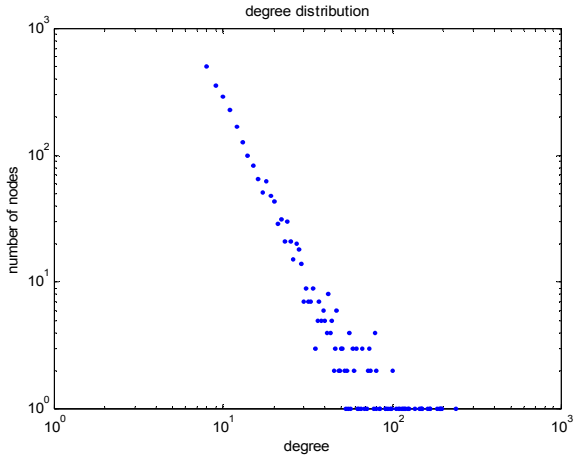

Figure 3 – Degree distribution of the free-scale network

3. Results and Discussion

In order to test the optimization efficiency and the algorithm behavior, in different topologies of the life space, we considered two benchmark problems widely used in evolutionary optimization: the Rastrigin's function (R) and the Ackley's function (A) [9]. Both are multimodal functions with a lot of local minima, which are likely to result in premature convergence. Their setups are the following:

$$\min \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right], \quad x_i \in [-5.12, 5.12],$$

for the minimization of the Rastrigin function and

$$\min \left[1 + e - \exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i) \right) \right],$$

$$x_i \in [-20, 30]$$

for the minimization of the Ackley's function. For each of them the optimal value of the cost function is 0 and corresponds to $x_i^o = 0, \quad i = 1, \ldots, n$.

We considered in our experimentation $n = 30$ for (R) and $n = 15$ for (A). The size of the network that represents the life space was set to 2500, which corresponded to a 50×50 square grid. The initial population size was set to 100, but this value is not controlled during the evolution by any external rule and the regime population size is an emergent characteristic of the environment. The only parameters we fixed for the individuals are the ones regulating their reproduction. In particular we chose for the maximum mutation amplitude as percentage of the search interval for each variable $\mu = 10\%$ for (R) and $\mu = 15\%$ for (A). We considered different realizations of

each topology and for each network we averaged the results of the algorithm over 10 runs with different random initial conditions, in order to have smooth curves.

To quantify the efficiency of the algorithm for the different topologies we considered the mean slope of the logarithm of the cost function when this is approaching 0, with respect to the number of function evaluations (table 1). In figures 4 and 5 the convergence of the algorithm over different topologies is plotted for (R) and (A) respectively.

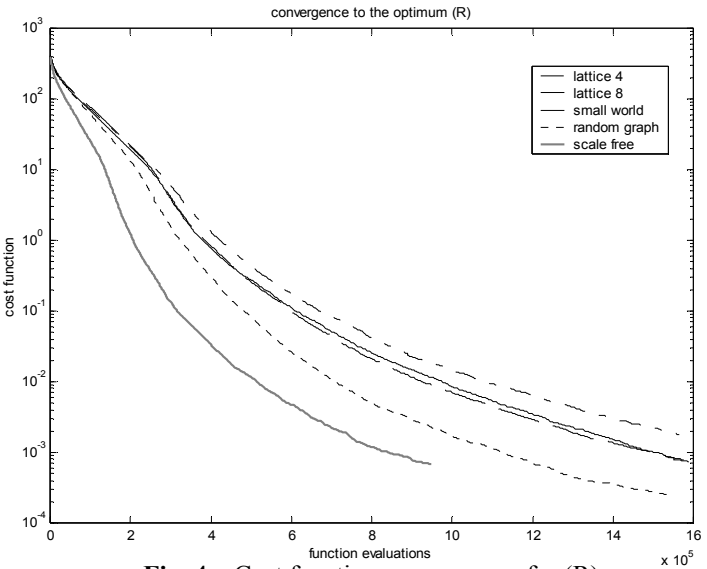

Fig. 4 – Cost function convergence for (R)

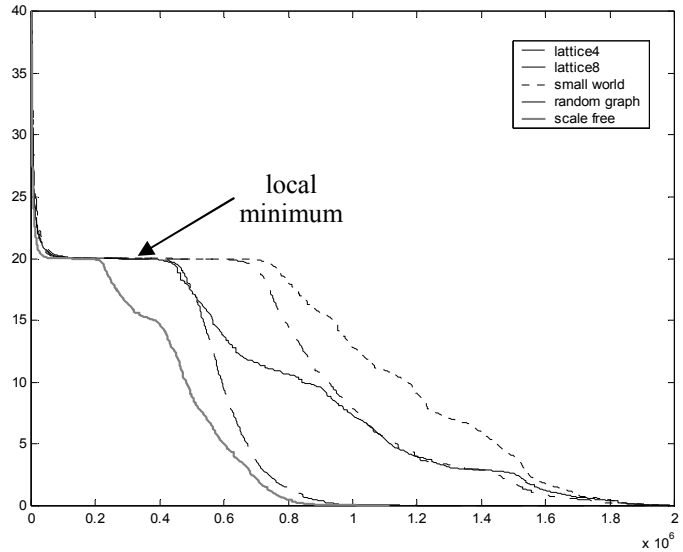

Figure 5 – Cost function convergence for (A)

For each of the functions we minimized, a remarkable improvement in optimum reaching is achieved for the considered alternative topologies. The most impressive results are achieved for the scale free space. This topology is very effective in exploiting the communication between the agents and, as we are going to show, it allows more interactions among them. As a consequence, the optimization is faster than for the other topologies and the environment results also in a higher mobility as can be seen by the shorter time to get out of a local minimum encountered in the minimization of the Ackley's function (see fig. 5). Both in (R) and in (A), we achieve for this topology an overall improvement that almost doubles the performance of the original algorithm (degree 4 lattice). The minor efficiency in (A) is due to a higher level of complexity in the fitness landscape than (R).

Average Slope (decades/(f_eval *10^6))	(R)	(A)
Lattice ($k = 4$, $p = 0$)	3.37	1.25
Lattice ($k = 8$, $p = 0$)	3.50	1.22
Small World ($p = 0.001$)	3.51	1.18
Random Graph ($p = 1$)	4.27	2.51
Scale-Free ($\lambda = 3$)	**5.79**	**2.95**

Table 1 – Efficiency over different topologies

As we said, using different topologies leads to different 'social policies' (interactions, reproductions and deaths) for the artificial life environment. In fig. 6 we compare the time series of the number of individuals for the degree 8 lattice, the random graph and the scale-free network for (R). As we mentioned above, the population size emerges as a property of the particular setup of the algorithm and the number of individuals oscillates around a constant value. One can notice the drastic reduction in this number in the scale-free topology. This reduces the computational cost per cycle and speeds up the optimization process.

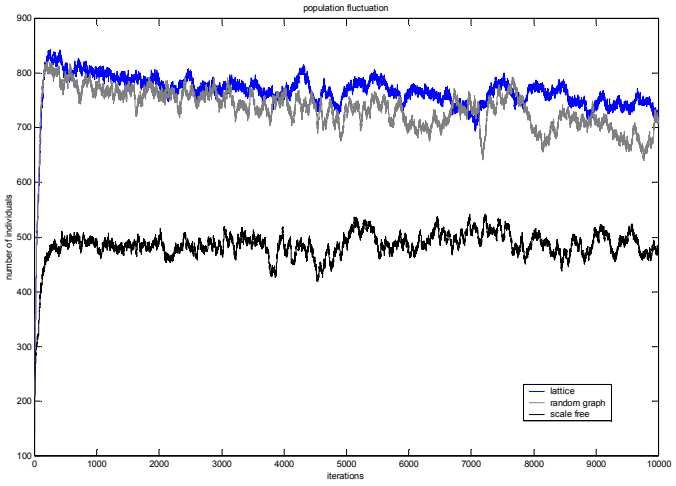

Figure 6 – Population fluctuations

The empirical explanation of this fact can be given considering another important issue, which we plot for (R) in fig. 7: the number of meetings as a percentage of the number of individuals. In a scale-free network there are cells more attractive than others are, because of their high degree (see fig. 3). In such a way several individuals are

likely to try to go in that site and in doing that they happen to meet other agents. As we see from this comparison, the percentage of meeting is not lowered by the minor number of individuals moving on a network with the same dimension of the other topologies and on the contrary it happens to be even higher than the others. These meetings create selection in the environment and many low-fit individuals are eliminated. The number of meetings, then, gives us also information about the density of population over the network. A large number of meetings means that a large number of individuals are visiting the same zone of the network. This reduces the probability to find an empty cell where to place one's son. In such a way the number of births, which is plotted in fig. 8 for the three representative networks, is reduced as well and as a consequence we have to do a smaller number of function evaluations.

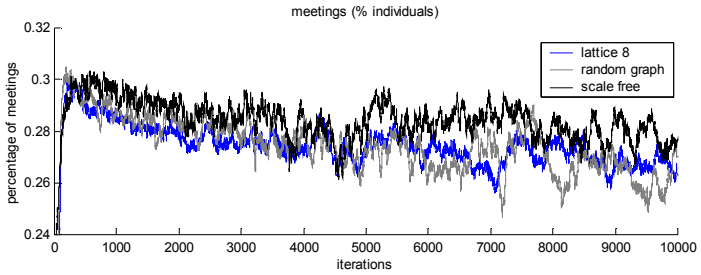

Figure 7 – Number of meetings

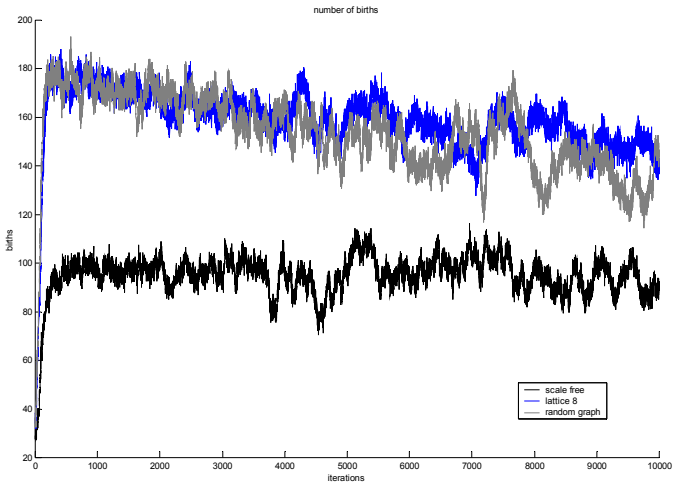

Figure 8 – Number of births

4. Conclusions

Different topologies for the life space of an artificial life environment have been studied. We tested the behavior of the algorithm in two well-known hard continuous optimization problems, when substituting the original lattice with other kinds of networks. In particular we began with the degree 4 lattice which was considered in the original implementation of the algorithm and first introduced a higher degree lattice, then a small world topology, a random graph and finally a scale-free network. Remarkable improvements in global optimum approaching are obtained in all of these cases but the best result is by far the one obtained considering the scale-free, which performed an overall efficiency more than the double with respect to the original lattice. Discussions and empirical observations about these first results are given.

Our choices in the implementation of the algorithm (in particular the ones regarding the mutation) led to optimization processes not as efficient as the state of the art, even if they are comparable, but our interest in this paper lied in a comparative study between different realizations of the same algorithm. Further efforts will be devoted in future to exploit these first achievements for more competitive implementations. In particular, in order to obtain results similar to the state of the art, we will probably have to consider a hybrid realization of our evolutionary algorithm to speed up the local search process, which now is performed with the same rules of the initial global one.

A better understanding of our results could come out from a deeper description of the individuals' dynamics in the environment. In particular we could examine the artificial life itself as an evolving complex network of interactions and study some statistical descriptors of such a network. Finally, a formal theoretical approach to this problem is also needed, in order to explain the improvement achieved.

Acknowledgements

This work was partially supported by the NATO Collaborative linkage grain PST.CLG.978512.

References

[1] Alba, E., Troya, J. M., *Cellular Evolutionary Algorithms: Evaluating the Influence of Ratio*, Proc. of PPSN 2000, pp. 29-38, 2000

[2] Annunziato, M., Bruni, C., Lucchetti, M., Pizzuti, S., *Artificial Life approach for Continuous Optimisation of non-stationary Dynamical Systems*, Integrated Computer-Aided Engineering, Vol. 10, n. 2, pp. 111-125, 2003

[3] Baluja, S., *Structure and Performance of Fine-Grain Parallelism in Genetic Search*, in Proc. of the 5th Int. Conf. on Genetic Algorithms, pp. 155-162, 1993

[4] Coven, R., Havlin, S., ben-Avraham, D., *Structural Properties of Scale-Free Networks*, chap. 4 in "Handbook of graphs and networks", Eds. S. Bornholdt and H. G. Schuster, Wiley-VCH, 2002

[5] Dorogotvsev, S. N., Mendes, J. F. F., *Evolution of Networks*, Advances in Physics, Vol. 51, n. 4, pp. 1079-1187, 2002

[6] Fleming, P. J., Purshouse, R. C., *Evolutionary algorithms in control systems engineering: a survey*, Control Engineering Practice, vol. 10, pp. 1223-1241, 2002

[7] Huang, S.-Y., Zou, X.-W., Tan, Z.-J., Jin, Z.-Z., *Network-induced nonequilibrium phase transition in the "game of Life"*, Phys. Rev. E, vol. 67, 2003

[8] Satoh, T., Kuwabara, H., Kanezashi, M., Nara, K., *Artificial life system and its application to multiple-fuel economic load dispatch problem*, Proc. of CEC 2002, vol. 2, pp. 1432-1437, 2002

[9] Yao, X., *Global Optimisation by Evolutionary Algorithms*, in Proc. of the 2nd Aizu International Symposium on Parallel Algorithms/Architecture Synthesis, pp. 17-21, 1997