

Dynamic and Partially Connected Ring Topologies for Evolutionary Algorithms with Structured Populations

C. M. Fernandes^{1,2}, J.L.J. Laredo³, J.J. Merelo², C. Cotta⁴, and A.C. Rosa¹

¹ LaSEEB-ISR-IST, University of Lisbon

² Department of Architecture and Computer Technology, Univ. of Granada, Spain

³ Faculty of Sciences, Technology and Communications, U. of Luxembourg

⁴ Departamento de Lenguajes y Ciencias de la Computación, U. of Malaga, Spain
{cfernandes,acrosa}@laseeb.org, juan.jimenez@uni.lu, jjmere-
lo@gmail.com, ccottap@lcc.uma.es

Abstract. This paper investigates dynamic and partially connected ring topologies for cellular Evolutionary Algorithms (cEA). We hypothesize that these structures maintain population diversity at a higher level and reduce the risk of premature convergence to local optima on deceptive, multimodal and NP-hard fitness landscapes. A general framework for modelling partially connected topologies is proposed and three different schemes are tested. The results show that the structures improve the rate of convergence to global optima when compared to cEAs with standard topologies (ring, rectangular and square) on quasi-deceptive, deceptive and NP-hard problems. Optimal population size tests demonstrate that the proposed topologies require smaller populations when compared to traditional cEAs.

1 Introduction

In standard Evolutionary Algorithms (EAs), all individuals are potential partners, i.e., there are no mating restrictions in the population preventing the pair-wise recombination of individuals. In genetics, this behavior is called *panmixia*, and the respective populations are called *panmictic*. For that reason, standard EAs without mating restrictions are also called *panmictic* EAs.

In panmictic EAs, genotypic representation, operators, selection schemes and population size are typical working mechanisms that require design choices. However, a population structure may be also introduced in the design scheme of EAs. The structure then specifies a network of acquaintances over which individuals can interact: mating or selection is restricted to neighborhoods within the network. The non-panmictic EAs that use this scheme are known as *spatially structured* EAs [12]. Spatially structured EAs include fine-grained approaches such as cellular EAs (cEAs) [1] and coarse-grained approaches such as island models [3]. In cEAs, the population is distributed in a grid and the interaction is restricted to the individuals' neighborhood. In island EAs, different subpopulations evolve isolated from each other and occasionally exchange individuals using a predefined strategy which specifies the rate and quantity of information to transfer.

The main disadvantage of island and cellular EAs is that their base-structures require extra designing and tuning effort. In addition, the chosen structure affects the

connectivity and the performance of the algorithm. In the case of island models, this added complexity translates in deciding policies for the migration frequency, selection and replacement of migrants and the topology itself. As for traditional cEAs, they use static structures that impose a rigid connectivity between the individuals. The investigation in this paper is an attempt to design a simple dynamic topology for cEAs, with a varying neighborhood degree and an intrinsic clustering behavior that approaches the cEA to an island model. In fact, the resulting structure may be considered a hybridization between a cellular and an island-based EA. This study is restricted to 1-dimensional structures, also known as *ring topologies*. The case of 2-dimensional population structures is left for a future investigation.

In the proposed partially connected ring topology the individuals are distributed in a 1-dimensional grid with size $1 \times Y$, where $1 \times Y > n$ and n is the population size. Therefore, there are $Y - n$ empty nodes or gaps in the network. Every time-step, each individual tries to recombine with one of its left or right neighbors (decided by tournament). If the individual has only one neighbor, it recombines with that neighbor. If there are no neighbors, there is no crossover and only mutation is applied. The structure is dynamic: in each time-step, every solution is allowed to move to neighboring nodes (if there are empty nodes in the individual's neighborhood).

With this scheme different niches may appear and disappear at run-time as the flow of information is interrupted by gaps. However, these gaps change during the run: the resulting cEA has certain resemblance with an island model, with dynamic clusters (or sub-populations) of individuals with varying size. We hypothesize that with this scheme the population diversity decreases at a lower rate (when compared to a standard ring topology), the optimal populations for a high rate of convergence are smaller, and the performance of the cEA on deceptive and hard problems is improved. The results of the experiments confirm the assumptions.

The remaining of the paper is structured as follows: Section 2 gives a background review on cEAs and on the effects of the topology on the diversity; Section 3 describes the proposed partially connected topologies; Section 4 describes the experiments and the results; Section 5 concludes the paper and outlines future lines of work.

2 Background Review

The initial objective of spatially structured EAs was to develop a framework for studying massive parallelization. However, the need to provide traditional EAs with a proper balance between exploration and exploitation motivated several lines of research that explore the potentiality of different population structures in maintaining genetic diversity [1]. The primary focus of the field has been on static regular lattices: every individual has a fixed number of potential interaction partners. Additionally, complex population structures have been studied (see [8] and [12]), many of them using recent developments in network theory.

In standard cEAs, the most typical population structure is a toroidal 2-dimensional grid with size $X \times Y$. The grid may be square or rectangular. The neighborhood of an individual is then defined according to a radius centered in the individual location. In this paper, we restrict the study to von Neumann neighborhood with radius 1, i.e., the

neighborhood of each individual consists of the individual itself and the individuals at adjacent North, East, South and West nodes. When the size of the grid is set to $1 \times Y$ (ring), the neighborhood consists of the individual and its left and right neighbors.

Standard cEAs have some drawbacks: synchronicity (in most cases) and a strong dependence on the problem, since the genetic diversity promoted by a prefixed topology is uncorrelated to the problem structure. In addition, the rigid connectivity of static structures may negatively affect the convergence abilities of the algorithms on some kind of problems, in which genetic diversity is crucial for escaping local optima.

For that reason, dynamic population structures have recently raised the interest of cEAs researchers. To the extent of our knowledge, only three works address explicitly the issue of dynamic population structures in cellular EAs. In 1, Alba and Dorronsoro dynamically change the ratio that defines the neighborhood of interaction. Since the ratio may affect selection pressure, the authors analyze the influence of its value on the balance between exploration and exploitation. However, the base-structure of the cellular EA is maintained throughout the run. In 12, Whitacre et al. focus on two important conditions missing in EA populations: a self-organized definition of locality and interaction epistasis. With that purpose in mind, they propose a dynamic structure and conclude that the two features, when combined, provide behaviors not present in the traditional spatially structured EAs. The most noticeable change is an unprecedented capacity for sustainable coexistence of genetically distinct individuals within a single population. The authors state that the capacity for sustained genetic diversity is not imposed on the population; instead, it emerges as a natural consequence of the dynamics of the system. Laredo et al. 7 proposed a framework for EAs based on peer-to-peer networks 10. Within a simulated network, they model the dynamics of real networks and conclude that their system is able to achieve better performance than traditional EAs on a wide range of problems, while being scalable and resilient to the volatility of nodes in the network.

In this paper we try a different approach. The radius of the neighborhood is fixed, and the typical grid structure is maintained. However, the size of the grid, which is usually set to $X \times Y = n$, is increased so that $X \times Y > n$ and some cells remain unoccupied. With empty cells in the grid, the individuals are then allowed to move to adjacent cells, according to a specific movement rule. Three different movement strategies are tested. The proposed scheme has been inspired by the work on a self-organized population of simple particles described in 2. Recently, a similar structure has been used for defining the interaction network of the Particle Swarm algorithm 6 with promising results 3. The following section describes the original system and its application to the particular case of the cEA.

3 Partially Connected Ring Topologies

As stated above, traditionally, cEAs are structured on 2-dimensional toroidal grids with size $X \times Y$, and the population size n is set to $n = X \times Y$. The main idea of this paper is to use populations structured in grids such that $n < X \times Y$. For that purpose, the dynamic complex system proposed by Fernandes et al. in 2 has been adapted for structuring populations.

-
1. Randomly place n particles in a grid of node with size $X \times Y$
 2. Randomly attribute a fitness value to each particle
 3. For each particle do
 4. check neighborhood for marks and other particles
 5. if no marks in the neighborhood
 6. move to a free node in the neighborhood (if any)
 7. if there are marks in the neighborhood
 8. move to the site of the nearest mark
 9. leave a mark in the previous site
 10. erase the mark in the new site
 11. if stop criteria not met return to 3
-

Figure 1. Pseudo-code of the original complex system 2.

The algorithm in 2 is a discrete complex adaptive system described by a set of local rules. These rules define the actions of a population of n simple particles that move on a 2-dimensional toroidal grid of nodes with size $X \times Y$. In each time-step, every particle tries to move to an adjacent node. The rules that control the particles' movements and the detailed description of the system are given below (please see also the pseudo-code in Figure 1).

At $t = 0$, the particles are assigned a random *fitness* value in the range $[0,1]$ and randomly distributed in a $X \times Y$ grid of nodes. Then, at each time-step, each particle moves to an adjacent free node (if any), leaving a mark with information on its current status in the previous node.

The particles decide where to go by inspecting their neighborhood. If there are no free nodes in the neighborhood (i.e., all the cells are occupied by particles), the particle stays in that same node until the next iteration. If it finds free nodes, the particle checks for marks. If it finds no marks, it just randomly chooses a destination node between the free neighboring nodes. If marks are found, the particle moves to the node with the most similar mark. Whenever a particle changes its position, it leaves a mark in its previous location. Furthermore, the marks only remain in the *habitat* for one iteration. Communicating, by depositing and following information, is the base-rule of the system.

This simple set of rules leads to a dynamic global behavior that displays signs of self-organization. A structure of particles, formed by clusters and paths, emerges on the habitat. However, these clusters are far from being static and, in a few generations, the distribution of the whole population may change dramatically (while maintaining a typical configuration of clusters and paths). The population's behavior is not ordered (nor chaotic).

The translation of this system to a population structure for cEAs can be straightforwardly done. For instance, the particles can be the individuals of the algorithm and the marks can be the fitness of the individuals. Moreover, other rules may be easily implemented and tested. In order to investigate the potentiality of partially connected grid topologies, three different movement rules have been used.

- 1) Fitness-based movement rule (f): as in the original model, the marks are the fitness of the individuals.
- 2) Similarity-based movement rule (s): the marks are the genotype of the individual that visited the node in the previous iteration; when deciding the destination node,

the individual computes the Hamming distance between its own genotype and the mark. Then, it moves to the node that minimizes the Hamming distance. (As in the original model, if there are no marks in the neighbourhood the individual chooses randomly an empty adjacent node).

- 3) Random movement rule (r): there are no marks and the individuals move to adjacent cells, select randomly amongst the empty ones.

Since the proposed population structure generates islands of individuals, we hypothesize that the genetic diversity of the population is maintained at a higher level (when compared to the standard ring topology). Therefore, exploration is increased and exploitation is performed at the local level by several subpopulations. Such characteristics could benefit the cEA when optimizing deceptive and multimodal hard problems. The results in the following section confirm these hypotheses.

The proposed structure and the three update schemes can be applied to the general case of 2-dimensional grid with size $X \times Y$. However, in this paper we restrict the study to the 1-dimensional case and compare the proposed structure to standard ring topologies. The 1-dimensional base-model display interesting properties, which are described in [2]. The system shows a mixture of order and randomness which is typical, for instance, of class 4 cellular automata [5]. Some clusters of particles move up or down, while free particles randomly move through the grid until they are “captured” by a cluster. Meanwhile, clusters disaggregate, freeing more “wandering” particles. The main goal and the motivation behind this work are to explore these emergent properties of the model, adding a self-organized dynamics to cEAs that may help them to escape more often from local optima traps.

The resulting cEA is described in Figure 2. Please note that the main differences to a standard cEA are that the grid size is larger than n and that when computing the neighborhood the algorithm may find two, one or zero potential partners, while in the

-
1. For each individual $i \leftarrow 1$ to n :
 - 1.1. Initialize individual i
 - 1.2. Evaluate individual i : $f(\bar{x}_i)$
 2. Set grid size: $X \times Y$: $X \times Y > n$
 3. Place the individuals randomly on the grid
 4. For each individual $i \leftarrow 1$ to n :
 - 4.1. Compute neighborhood
 - 4.2. Parent 1 is individual i
 - 4.3. Parent 2 is the best of the neighbors
 - 4.3 Crossover (parent 1, parent 2)
 - 4.4. Select randomly one of the offspring: offspring i
 - 4.5. Mutation (offspring i)
 - 4.6. Evaluate offspring i : $f(\bar{x}'_i)$
 - 4.6. Insert offspring i in temporary population P_t
 5. For each individual $i \leftarrow 1$ to n :
 - 5.1. Replace individual i by offspring i if $f(\bar{x}'_i) > f(\bar{x}_i)$ (maximization problems)
 - 5.1. Compute empty adjacent nodes.
 - 5.2. If at least one empty node, select destination node using movement rule.
 6. If the stop criterion is not met, go to 4
-

Figure 2. cEA on a partially connected grid.

standard ring topology an individual has always two potential partners for recombination. The following section tests the structures on a set of problems with deceptive landscapes and other characteristics that make them hard for standard EAs to solve.

4 Test Set and Results

In order to investigate their performance, the proposed partially connected ring topologies have been tested on trap functions with increasing degree of difficulty. The results were then compared to the standard square, rectangular and ring structures.

A trap function is a piecewise-linear function defined on *unitation* (the number of ones in a binary string) that has two distinct regions in the search space, one leading to the global optimum and the other leading to a local optimum. Depending on its parameters, trap functions may be deceptive or not. The trap functions in these experiments are defined by:

$$F(\vec{x}) = \begin{cases} k, & \text{if } u(\vec{x}) = k \\ k - 1 - u(\vec{x}), & \text{otherwise} \end{cases} \quad (1)$$

where $u(\vec{x})$ is the unitation function and k is the problem size (and also the fitness of the global optimum). With these definitions, order-3 traps are in the region between deceptive and non-deceptive, while order-2 are non-deceptive and order-4 are fully deceptive. Under these settings, it is possible to investigate not only how the algorithms scale on order- k trap functions but also to observe how that performance varies when moving from non-deceptive to deceptive search spaces. For that purpose, l -bit decomposable functions are constructed by juxtaposing m trap functions and summing the fitness of each sub-function to obtain the total fitness, obtaining the so-called m - k trap problems. Then, by increasing m it is possible to investigate how an algorithm scales.

In the first experiments, order-2, -3 and -4 trap functions were constructed by juxtaposing, respectively, 250, 125 and 75 subproblems, generating 500- (2-trap), 375- (3-trap) and 300-bit (4-trap) problems.

All the cEAs used in the experiments are synchronous (i.e., the offspring are placed in a temporal population and replacement is done after every individual generates one child). Parameterization was done after [1]: population size was set to 400; the recombination operator is the double point crossover with $p_c = 1.0$; mutation is bit-flip with $p_m = 1/l$, where l is the chromosome length. Only one offspring is placed in the temporal population (randomly chosen from the set of two children). In the replacement stage, the offspring replaces its parent if it's better.

The stop criteria are: to find the global optimum or to achieve a maximum of 3,000,000 function evaluations. The number of iterations required to meet the best solution is recorded and averaged over 50 runs. A success measure (successful runs) is defined as the number of runs in which an algorithm attains the global optimum.

Please note that the tests are not intended to show that the proposed structure is better than the standard ring topologies in a wide range of problems. We are first interested in understanding the behaviour of the partially connected rings, in general, and their performance on quasi-deceptive and deceptive problems, in particular.

In the proposed topologies, the empty nodes are obstacles for the flow of information through the population, which means that the search is performed by several subpopulations, although highly dynamic. It is expected therefore that the increase in exploration slows down the convergence speed of the algorithms. However, we expect the payoff to be to an increasing robustness, with the partially connected topologies being able to find the global optimum more often.

The results of the experiments on trap functions are shown in Table 1. The first relevant result is that the standard ring topology (1×400) outperforms the other static structures, not only on deceptive and quasi-deceptive functions, but also on the non-deceptive 2-trap function, finding the global optimum in every run. In this function, the partially connected topologies — random (r), fitness-based (f) and similarity-based (s) — also find the optimum in every run. However, they converge more slowly, probably due to their own balance between exploration and exploitation, which favours exploration (when compared to the standard ring). In 3-trap and 4-trap functions the partially connected rings are also slower (in general, they require about 10% more evaluations to reach the optimum), but in this case they converge more often to the global optimum. As expected, the empty nodes in the ring slow down the convergence speed but increase the convergence probability.

In the previous tests, the grid size of the partially connected topologies was set to 1×500 . It is expected that the size affects the speed and the convergence rate of the algorithm. A sparser structure increases exploration (at the expenses of convergence speed); with higher exploration the algorithm converges more often to the optimum.

Table 1 shows the performance of the partially connected rings with different ratios between population size and grid size. The population is set to 400 and the grid size is varied from 450 to 800. As in the previous experiments, the stop criteria are reaching the global optimum or 3,000,000 function evaluations. As expected, convergence speed decreases when the grid size is larger. But the number of successful runs also increases with the size of the structure. Increasing exploration slows down the search process but improves the success in reaching the optimum. By adjusting the size, it is possible to balance global and local search.

Table 1. Averaged function evaluations to a solution (AES), successful runs (SR) and averaged best fitness (FIT).

		20×20	10×40	1×400	$1 \times 500 (r)$	$1 \times 500 (f)$	$1 \times 500 (s)$
2-trap $l = 500$	AES	1084612.8 ± 380406.03	892547.4 ± 513403.61	567944.0 ± 40322.30	620104.0 ± 42747.69	608922.4 ± 44761.51	622936.0 ± 56503.71
	SR	(47)	(38)	(50)	(50)	(50)	(50)
	FIT	0.08 ± 0.34	0.24 ± 0.43	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
3-trap $l = 375$	AES	-	161066.7 ± 21289.96	697671.8 ± 108269.7	752178.7 ± 81058.45	752853.3 ± 83956.6	738248.9 ± 146850.3
	SR	(0)	(3)	(39)	(49)	(50)	(47)
	FIT	5.86 ± 2.53	3.60 ± 1.94	0.04 ± 0.47	0.02 ± 0.14	0.00 ± 0.00	0.06 ± 0.24
4-trap $l = 300$	AES	-	-	773238.7 ± 93423.95	842491.0 ± 121666.5	845810.5 ± 91364.75	869936.8 ± 126965.1
	SR	(0)	(0)	(31)	(35)	(38)	(43)
	FIT	6.56 ± 2.09	5.06 ± 2.10	0.42 ± 0.57	0.38 ± 0.66	0.30 ± 0.57	0.16 ± 0.42

Table 2. Order-4 trap functions. Varying the size of the grid. $n = 400$

	r	f	s
1×450	835837.8 ± 146794.4 (37)	774148.6 ± 112251.3 (35)	868514.3 ± 214231.8 (35)
1×500	842491.0 ± 121666.5 (35)	845810.5 ± 139281.9 (38)	869936.8 ± 154717.9 (38)
1×600	978195.1 ± 108800.1 (41)	984120.0 ± 183584.7 (40)	959930.0 ± 110843.9 (40)
1×700	1122688.4 ± 132330.9 (43)	1126234.2 ± 147857.3 (41)	1160488.9 ± 176815.8 (45)
1×800	1327351.1 ± 118173.3 (45)	1343266.7 ± 144683.4 (42)	1301502.2 ± 127471.9 (45)

Finding the optimal population size for a given problem is a fundamental step when optimizing the performance of a given EA. In order to investigate the optimal population sizes for the different structures, we have used a selecto-recombinative version of the cEAs (i.e., without mutation) and the bisection method 9 (please note the bi-section method requires EAs without mutation).

The bisection method, described in Figure 3, is a simple but effective technique used to determine the optimal population size of selectorecombinative EAs. For this particular case the threshold T was set to 0.1 and initial population size was set to 50. Every configuration was run for 30 times before updating and the convergence criteria is met if 29 of those 30 runs converge towards the global optimum. The algorithms were tested with $p_c = 1.0$. Mutation probability was set to 0. After determining the optimal population size, the configuration with that n value was executed for 50 times and the number of evaluations necessary to reach the optimum was averaged over the successful runs. The results (optimal population size and averaged evaluations to a solution with that particular size) are given in Table 3.

-
1. Start with small n
 2. Double n until GA convergence criteria is met
 3. $(\min, \max) = (n/2, n)$
 4. repeat until $(\max - \min) / \min < T$
 $n = (\min + \max) / 2$
if n leads to convergence criteria
then $\max = n$
else $\min = n$
 5. Compute the statistics for this problem size using population size = \max
-

Figure 3. The bisection method for determining the optimal population size of a GA.

Table 3. Optimal population size and averaged evaluations to a solution.

		<i>ring</i>	<i>p. c. ring (r)</i>	<i>p. c. ring (f)</i>	<i>p. c. ring (s)</i>
4-trap $l = 52$	<i>n</i>	200	175	175	175
	AES	29376.0 ± 6235.7	30183.7 ± 5422.3	29207.2 ± 6796.6	29830.1 ± 6973.7
4-trap $l = 100$	<i>n</i>	350	300	275	300
	AES	111271.4 ± 15166.4	100383.2 ± 12045.6	93848.3 ± 13075.7	100306.6 ± 11337.6
4-trap $l = 200$	<i>n</i>	600	500	500	500
	AES	401706.1 ± 44220.1	357428.4 ± 29096.9	369945.3 ± 45849.2	367833.3 ± 39049.3

The main conclusions are that, as expected, the partially connected topologies require smaller populations than the fully connected ring. In the case of the quasi-deceptive and deceptive functions, smaller populations lead to faster algorithms. Therefore, and according to the results in Tables 2 and 3, we conclude that the proposed topologies are more robust, although slower, when the population size is set to the same value, and faster when the population size is set to a size that assures a convergence rate close to 100%.

A final set of experiments aims at comparing the standard ring cEA with the random movement version of the proposed ring topology on a wider set of problems. For that purpose, MMDP and Trident problems have been added to the test set. Trident functions are *needle in the haystack* problems that exploit the ability of EAs to mix good but significantly different solutions. The fitness function of the Trident used in this work has two components, *base* and *contribution*: $F(\vec{x}) = \text{base}(\vec{x}) + \text{contribution}(\vec{x})$. The base depends on unitation and is described by:

$$\text{base}(\vec{x}) = \|2 \cdot \mathbf{u}(\vec{x}) - \mathbf{l}\| \quad (4)$$

where l is the chromosome length. The contribution rewards certain configurations of strings that an equal number of 0's and 1's.

Table 4. MMDP. Contribution of each subproblem configuration to the fitness value.

$\mathbf{u}(\vec{x})$	0	1	2	3	4	5	6
$F(\vec{x})$	1.000000	0.000000	0.360384	0.640576	0.360384	0.000000	1.000000

Let L be the first half of the binary string x of length l and R the second half. The *contribution* is described by Equation 5:

$$\text{contribution}(\vec{x}) = \begin{cases} 2 \cdot l, & R = \bar{R} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where \bar{R} is the bitwise negation of R . The Trident accepts strings of length $2k$, where $k \geq 2$. For this paper, 64-bit strings were used.

The MMDP is an NP-hard problem that has been designed to be difficult for EAs. Like the trap functions with order ≥ 3 , MMDP is deceptive, but it is also multimodal. It consists of k 6-bits subproblems with two global optima and a deceptive attractor in

Table 5. Selecto-recombinative standard ring topology cEA and partially connected ring cEA with random movement. Optimal population size, average evaluations to a solution and Kolmogorov-Smirnov statistical tests with 0.05 level of significance.

	2-trap ($l = 400$)	3-trap ($l = 300$)	4-trap ($l = 200$)	MMDP ($l = 240$)	Trident ($l = 64$)
$1 \times n$	$n = 500$ 455620.7 ± 45848.64	$n = 600$ 524524.1 ± 80506.1	$n = 550$ 372881.0 ± 53921.9	$n = 750$ 520551.7 ± 47646.71	$n = 350$ 91374.14 ± 36194.35
$1 \times (1.5n)$	$n = 450$ 444171.8 ± 32089.3 (+~)	$n = 500$ 427207.6 ± 34671.3 (++)	$n = 475$ 350769.8 ± 36278.0 (++)	$n = 600$ 466453.1 ± 56597.55 (++)	$n = 300$ 78352.55 ± 19654.27 (+~)

the middle of the fitness landscape. Each subproblem fitness values depend on the uninitiation function. Table 4 shows the contribution of each subproblem to the fitness value of a string. For the experiments, 240-bit strings were used.

The first test determines the optimal population size of each algorithm for each problem using the bi-section method. The standard ring is compared to the partially connected version with random selection of destination nodes. The results are in Table 5. Statistical tests (Kolmogorov-Smirnov statistical tests with 0.05 level of significance) that compare the AES of each algorithm in each function are also given: (+~) means that the partially connected ring is faster than the standard ring but the differences in the AES values are not statistically significant; (++) means that the partially connected ring is better and the differences are statistically significant.

The proposed topology outperforms the AES of the standard cEA in every function. The differences are statistically significant in order-3 and -4 traps and in the MMDP. The algorithm seems to be particularly suited for deceptive problems. In the second experiment, the cEAs are provided with mutation: mutation probability was set to $1/l$ for every test. The population size of both strategies is set to half of the optimal population size of the standard cEA, in order to investigate how the cEAs behave when the supply of raw building blocks is reduced and part of the genetic diversity is assured by mutation. (The bisection method determines the optimal population size for the selecto-recombinative version of the algorithm. When using mutation, that minimal population, which guarantees a high rate of convergence, may be reduced.) Results are in Table 6.

Table 6. Standard ring topology cEA and partially connected ring cEA with random movement. Evaluations to a solution, successful runs and best fitness.

		2-trap	3-trap	4-trap	MMDP	Trident
$1 \times n$	AES	306562.5 ± 56485.3	424021.6 ± 77027.02	351001.9 ± 67053.15	610437.5 ± 408323.2	24027.5 ± 5782.976
	SR	(48)	(37)	(27)	(36)	(50)
	FIT	0.40 \pm 0.20	0.260 \pm 0.44	0.60 \pm 0.75	0.11 \pm 0.18	0.0\pm0.0
$1 \times (1.5n)$	AES	352105.0 ± 37759.4	530622 ± 87469.522	446441.1 ± 89766.2	638250.0 ± 243028.4	27895.0 ± 6424.288
	SR	(50)	(50)	(45)	(45)	(50)
	FIT	0.0\pm0.0	0.0\pm0.0	0.12\pm0.38	0.04\pm0.12	0.0\pm0.0

With these settings, the success rates of the standard ring are significantly reduced in the deceptive problems, while the partially connected structure attains success rates above 90% in every problem. In the 2-trap and Trident functions the results are similar: there are no statistical differences between the AES values.

A final note on the implementation of the proposed algorithm: Although uniprocessor implementations are common, cEAs have been initially conceived for parallel computing frameworks, in which several processors are structured in a static grid or ring topology. The proposed schemes could model some properties of networks of processors (such as fail or delays in the communication, represented here by empty cells), but they may be hard to implement in a multiprocessor framework. It is necessary to devise a probability-based partially connected ring topology, where the size of the ring is maintained and links between the nodes are connected and disconnected according to probability values, adjacency rules or even self-organized properties. The results described in this section, which show that the proposed partially connected rings for cEAs are able to improve standard structures in hard problems with deceptive landscapes, are promising and motivate future research on alternative models of the proposed scheme that do not require empty nodes in the network.

5 Conclusions and Future Work

This paper describes a partially connected 1-dimensional cellular Evolutionary Algorithm (cEA). The structure consists of a population of n individuals randomly distributed in a grid with size $1 \times Y$, where $Y > n$. In every time-step, the individuals try to move to adjacent nodes, according to specific rules. The resulting structure displays an island-model behaviour that promotes genetic diversity and reduces the minimum population size that assures a high rate of convergence to a global optimum.

Three movement rules have been tested: random, fitness based and similarity based rules. The results of the different schemes are similar and further investigation is required in order to understand the potential of each one. The most important outcome here is that the partially connected structure significantly improves the success rates of the standard structure on quasi-deceptive and deceptive problems. Optimal population size tests with selecto-recombinative cEAs show that the proposed algorithm requires smaller populations for attaining the optimum, which means that it has a better ability to recombine the raw building-blocks provided by the initial population and maintain genetic diversity.

Two main lines of research are planned for the future. First, we will investigate the behaviour of general 2-dimensional partially connected grids and compare it to square and rectangular static topologies. The second line of research is dedicated to modelling the partially connected rings in a probability-based model, without empty nodes between the individuals. This way, a multiprocessor approach may be implemented.

Acknowledgements. The first author wishes to thank FCT, *Ministério da Ciência e Tecnologia*, his Research Fellowship SFRH/BPD/66876/2009. The work was supported by FCT PROJECT [PEst-OE/EEI/LA0009/2013], Spanish Ministry of Science and Innovation project TIN2011-28627-C04-02, Andalusian Regional Government P08-TIC-03903, CEI-BioTIC UGR project CEI2013-P-14, and UL-EvoPerf project.

References

1. Alba, E., Dorronsoro, B.: The exploration/exploitation tradeoff in dynamic cellular genetic algorithms, *IEEE Transactions Evolutionary Computation* 9, pp. 126–142, 2005.
2. Fernandes, C.M., Laredo, J.L.L., Merelo, J.J., Cotta, C., Rosa, A.C.: Towards a 2-dimensional Framework for Structured Population-based Metaheuristics, in *Proc. of IEEE International Conference on Complex Systems*, pp. 1-6, 2012.
3. Fernandes, C.M., Laredo, J.L.L., Merelo, J.J., Cotta, C., Rosa, A.C.: A Study on Time-Varying Partially Connected Topologies for the Particle Swarm, in *Proc. of the IEEE Congress on Evolutionary Computation, IEEE*, pp. 2450-2456, 2013.
4. Gordon, V., Whitley, L.: Serial and Parallel Genetic Algorithms as Function Optimizers, in *Proc. 5th ICGA*, 00. 177-183, 1993.
5. Ilachinski, A.; *Cellular Automata: A Discrete Universe*, World Scientific, 2001.
6. Kennedy, J., Eberhart, R.C.: *Swarm Intelligence*, Morgan Kaufmann, San Francisco, 2001.
7. Laredo, J.L.J., Castillo, P.A., Mora, A.M. Merelo, J.J., Fernandes, C.M., Resilience to churn of a peer-to-peer evolutionary algorithm, *International Journal of High Performance Systems Architecture*, 1(4), pp. 260–268, 2008.
8. Payne, J.L. Eppstein, M.J.: Emergent mating topologies in spatially structured genetic algorithms, in *Proc. 8th GECCO*, 207-214, 2006.
9. Sastry, K.: Evaluation-relaxation schemes for Genetic and Evolutionary Algorithms. Msc Thesis, University of Illinois, Urbana, IL, USA, (2001)
10. Steinmetz R., Wehrle, K., Eds., *Peer-to-Peer Systems and Applications*, Lecture Notes in Computer Science, vol. 3485, Springer, 2005.
11. Tomassini, M.: *Spatially Structured Evolutionary Algorithms*, Springer, Heidelberg, 2005.
12. Whitacre, J.M., Sarker, R.A., Pham, Q.: The self-organization of interaction networks for nature-inspired optimization, *IEEE Transactions on Evolutionary Computation*, 12, pp. 220–230, 2008.