Internal Lattice Reconfiguration for Diversity Tuning in Cellular Genetic Algorithms

Alicia Morales-Reyes[1,2]*, **Ahmet T. Erdogan**[1]

[1] School of Engineering, The University of Edinburgh, Edinburgh, Scotland, United Kingdom, [2] Computer Science Department, Instituto Nacional de Astrofisica, Optica y Electronica, Puebla, Mexico

Abstract

Cellular Genetic Algorithms (cGAs) have attracted the attention of researchers due to their high performance, ease of implementation and massive parallelism. Maintaining an adequate balance between exploitative and explorative search is essential when studying evolutionary optimization techniques. In this respect, cGAs inherently possess a number of structural configuration parameters that are able to sustain diversity during evolution. In this study, the internal reconfiguration of the lattice is proposed to constantly or adaptively control the exploration-exploitation trade-off. Genetic operators are characterized in their simplest form since algorithmic performance is assessed on implemented reconfiguration mechanisms. Moreover, *internal* reconfiguration allows the adjacency of individuals to be maintained. Hence, any improvement in performance is only a consequence of topological changes. Two local selection methods presenting opposite selection pressures are used in order to evaluate the influence of the proposed techniques. Problems ranging from continuous to real world and combinatorial are tackled. Empirical results are supported statistically in terms of efficiency and efficacy.

Citation: Morales-Reyes A, Erdogan AT (2012) Internal Lattice Reconfiguration for Diversity Tuning in Cellular Genetic Algorithms. PLoS ONE 7(7): e41279. doi:10.1371/journal.pone.0041279

Editor: Josh Bongard, University of Vermont, United States of America

Received February 9, 2012; **Accepted** June 19, 2012; **Published** July 25, 2012

Funding: Alicia Morales-Reyes is supported by the Mexican National Council for Science and Technology (www.conacyt.mx) studentship number 168016 and the National Institute for Astrophysics Optics and Electronics (www.inaoep.mx). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: a.morales@inaoep.mx

Introduction

Evolutionary Algorithms (EAs) are stochastic search techniques based on the principles of evolution. Amongst EAs, Genetic Algorithms (GAs) provide a competitive approach that has been successfully applied to a wide variety of difficult optimization problems [1]. Parallel Genetic Algorithms (PGAs) are classified by their grain in coarse/distributed and fine/cellular GAs [2,3]. Coarse PGAs consist of several independent homogeneous or heterogeneous sub-populations evolving in parallel with individuals migrating among them. Fine or cellular GAs consist of a decentralized population where individuals interact with others located at nearby positions.

From the theoretical to the real world and particularly in the hardware implementation arena, GAs feasibility has been investigated. From the Very Large Scale Integration (VLSI) point of view, GAs possess important characteristics that makes them suitable for hardware implementation targeting real time applications. Parallelized versions of GAs have been used as optimizers and adapted for hardware implementation targeting real-time performance. Among problems solved by parallel GAs are: the image registration problem [4–6], the disc scheduling problem [7] and the GPS attitude parameters determination problem [8–11].

In [7], a cGA architecture was developed targeting the disc scheduling problem. It consists in finding the best attending tasks order to reduce the access time per request. In this architecture, a cellular GA encodes up to 32 queued requests ordered by a fitness function that minimizes latency and search time per request. The architecture achieved an scheduling time of up to 2 milliseconds per access request and 4 milliseconds of searching time per request. Thus, timing constraints were fulfilled. Ordered based crossover and mutation operations were performed. Ordered based crossover randomly selects chromosome positions in one parent, finds these positions in the second parent and copies corresponding alleles to the offspring. Thereafter, a reordering policy is applied, for selected genes to maintain their positions in the offspring. Ordered based mutation selects certain chromosome positions and mutate the alleles.

An image processing architecture was developed to tackle the image registration problem that presents real-time constraints [4,5]. First, in [4] an architecture for image registration using a cellular GA was proposed. The image registration problem consists in matching a 2-D captured image to a reference image. A transformation between both images is necessary. The fitness function measures the number of corresponding pixels between captured and reference images. Once a perfect match has been found, the transformation encodes the object's position and orientation. Each transformation parameters were encoded in 6 bits. At algorithmic level, an extra step was added to the canonical cGA procedure defined by Tomassini [12]. Each transformation parameter is incremented or decremented in one unit for promotion through hill climbing, while the best individual is always kept. Results showed that real-time constraints are fulfilled up to a tenth of a second to match the images, corresponding approximately to 35 generations for 64×64 images.

In [5], an improved cGA architecture for image processing combining data compression and image registration is presented. For the image registration, a cGA determines an affine transformation for the image coordinates with respect to a reference image. The fitness function minimizes the error between the transformed and the referenced images. Then, the Discrete Cosine Transform (DCT) using a data compression algorithm is applied in order to reduce the memory storage requirements. In terms of speed, processing one chromosome took up to 2 milliseconds with limited accuracy for 64×64 images.

A cellular GA architecture tackling the GPS attitude determination problem was developed by Xu et al. In [8,9] a method to determine the attitude parameters of a vehicle based on the Global Positioning System (GPS) technology and the Ambiguity Function Method (AFM), is presented. The attitude parameters are determined by the vector difference between GPS receivers attached to the vehicle. Binary chromosome representation of 32 bits is used. The azimuth angle is encoded in the first 14 bits, the elevation angle in the next 10 bits and the base line length in the last 8 bits. In [9] a comparison among a panmictic GA, a panmictic GA with fine and coarse search stages, and a cellular GA with fine and coarse stages was carried out. Different array sizes were tested in order to find the best population size to solve the problem. Algorithmic performances were measured in terms of the average number of generations, and the hit rate together with the results accuracy.

Stefatos et al. extended the GPS attitude architecture presenting a novel approach from a fault tolerance perspective to deal with Single Hard Errors (SHEs) and implicitly with Single Event Upsets (SEUs), both are Single Event Effects (SEEs) subclasses [10]. Erroneous bit flipping in data registers, temporary (SEUs) or permanently (SHEs) is the effect of this kind of faults. If faulty registers correspond to critical data, the system's functionality fails. In the proposed architecture, a faulty scenario considering stuck at zero faults at fitness score registers was targeted. In the cGA, individuals with faulty fitness scores are not selected and therefore their solution is not spread throughout the population.

The fault tolerant architecture is a double layer approach, each layer implements a cellular GA. A computational layer to determine the GPS attitude parameters and a control layer to determine the best configuration of individuals or Processor Elements (PEs) to overcome a faulty scenario. Several experiments were carried out, first only the computational layer was tested in order to observe how the cGA deals with the faults. Then the control layer was executed and a performance comparison is carried out. Results showed a significant improvement in the system's performance when the control layer was active, in the worst case scenarios with 30% and 40% faulty PEs. Although, using the control layer allows an improvement in the system performance, there is a significant increment in the use of hardware resources. Another disadvantage is that the control layer was considered free of faults.

In [11] a high performance hardware architecture for the GPS attitude determination problem was presented; emphasis was paid to the speed, power consumption and hardware usage. Thus, a Coordinate Rotation Digital Computer (CORDIC) module was implemented to calculate the trigonometric functions required by the fitness function [13]. Results fulfil real-time constraints due to the simplified arithmetic units.

This article aims to provide a deeper insight into the great flexibility which cellular GAs inherently possess. When appropriately exploited, cGAs' unique properties as regards the interaction of individuals enhance their searching abilities. In EAs, the main interest is to preserve population diversity through satisfactorily balancing the exploration and exploitation of solutions throughout the search space; in order to successfully converge to the global optimum in the minimum number of generations [14]. In this regard, structural properties of cGAs have shown their effectiveness in maintaining this balance, in several cases improving cGAs performance [15]. In this study, a dynamic cellular approach that *internally* reconfigures a traditional square topology into smaller sub-cellular structures is investigated.

Initial studies on Genetic Algorithms were mainly based on panmictic populations [2]. However, evident disposition to parallelization expanded GAs horizons, opening new areas for investigation, such as detailed by Cantú-Paz in his study on parallel GAs approaches in [16,17]. In [16], parallel distributed GAs are analysed and new techniques are proposed to calculate population size according to problem's difficulty. Moreover, migration is investigated as a crucial genetic operator and as a possible reason for PGAs speed-ups [18].

In order to validate the importance of decentralized GAs, authors compared their results to those of standard GAs, and frequently referred to modified or improved versions of standard GAs, such as steady state GA or generational GA. In several cases, cGAs were proved to perform better in terms of algorithmic efficiency and efficacy [19]. Moreover, the effect of synchronism in migration policies of distributed approaches using cGAs or panmictic independent populations, has been studied and compared. Asynchronous cGAs proved to be a competitive option that outperform synchronous ones, and in several cases also outperform panmictic distributed GAs in terms of real time [20]. On the other hand, simplicity in implementation of processor elements is also an advantage that has been investigated and applied [11].

Alba et al. have widely studied decentralized GAs [21]. An important area covered by their work is to compare distributed parallel GAs considering homogeneous or heterogeneous populations or a mixture of these. In [22,23] a detailed analysis of several approaches at algorithmic and implementation levels is presented. Configurations of homogeneous sub-populations (only panmictic or cellular), heterogeneous sub-populations (a mixture of panmictic and cellular), comprising several migration and replacement criteria are evaluated. In general, their results showed the superiority of distributed cellular GAs in terms of efficacy over panmictic distributed PGAs. However, cellular GAs showed slower convergence times than standard steady-state distributed GAs. Communication among sub-populations was also assessed. Asynchronous communication leads to faster convergence and better speed-ups than synchronous ones.

In cellular GAs, the population is normally distributed in a torus like grid structure with wraparound edges. One individual is placed in each grid position and can only interact with nearby neighbours. Several parameters must be configured in order to reach the best algorithm performance, such as, shape and size of local neighbourhood and population topology, migration rate and frequency (in case explicit migration occurs), local selection and replacement policies, among others [24].

Initially, the study of cGAs has been developed following empirical knowledge through manually tuning and configuring the parameters mentioned above and through considering specific areas of application. However, there has also been an effort to provide a mathematical basis to explain their behaviour. For example, in order to provide a numerical relationship between grid structure and local neighbourhood, the dispersion of a points pattern has been defined as an appropriate measure known as the Neighbourhood to Grid Ratio (NGR) [15,25], more details are provided in the next section. It has been demonstrated that by having different neighbourhood sizes and shapes in combination with, for example a square topology, different selection pressures are obtained. Selection pressure in cGAs is measured as the number of generations it takes for the best individual of an initial population to spread its solution throughout the grid. This measure is carried out by using only local selection. Shorter take-over times indicate high selection pressures and vice versa. A more detailed analysis will be carried out in the next section.

This article aims to investigate how selection pressure can be constantly or adaptively controlled in order to improve the performance of cellular GAs, as a result of the population topology and the local neighbourhood configuration. Previously, researchers have proposed dynamically reconfiguring the population topology [26]. However, a form of migration occurs during the relocation of individuals; this approach from now on is named as *external* lattice reconfiguration. Three main scenarios were evaluated: static (fixed topology shape), pre-programmed (reconfiguration at a predefined time slot) and adaptive (dynamic reconfiguration) criteria. Statistically significant results were obtained for several difficult benchmark problems. The mapping of individuals, after grid re-arrangement, induces a loss of the natural adjacency of individuals, due to toroidal connection. In this research, adjacency among individuals is maintained while constant or adaptive lattice reconfiguration is carried out by subdividing the entire population into smaller square, rectangular or linear toroidal arrays, with no induced migration among them apart from neighbourhoods overlapping. Proposed *internal* reconfiguration mechanisms are applied to a set of continuous, real and combinatorial problems covering characteristics such as multimodality, epistasis, deceptiveness and non-regularity.

In the following section, a detailed explanation of the effect of controlling selection pressure through structural properties in cellular GAs is presented, together with proposed *internal* reconfiguration mechanisms.

Methods

Selection Pressure

Mechanisms proposed in this article control, constantly or adaptively, the induced selection pressure during evolution through the *internal* reconfiguration of the population topology. *Internal* lattice reconfiguration means that there is no alteration to the natural adjacency of individuals. Measuring selection pressure in cellular GAs has been fundamental and has allowed a more accurate study of their behaviour. Selection pressure in cGAs, is measured through the local selection of individuals, without intervention from other genetic operators; and it is empirically calculated as the number of generations the best individual of an initial population needs in order to spread its solution throughout the entire grid, concept also known as take-over time [12,15].

Previously, Dorronsoro et al. proposed several constant and adaptive criteria to *externally* modify the induced selection pressure in order to improve the performance of cGAs [26]. However, their approach considers an inherent migration mechanism; through it, new positions are calculated for individuals when the population topology is changed among square, rectangular and narrow lattice shapes. Although results showed an improvement in performance for several benchmark problems, it is not clear if this improvement is purely due to controlling the NG ratio or if it is also a consequence of inherent migration. In this article, the proposal is to internally reconfigure the population's topology, constantly or adaptively, while maintaining the original adjacency among individuals.

Selection pressure in cGAs provides important information about the evolutionary process showing levels of exploration or exploitation that certain population topology contribute to the search. In [26], diversity changes are measured as speed variations of the average population fitness score or the population entropy. If the average fitness increases drastically from one generation to the other, the search process proceeds rapidly and therefore exploitation is high. On the other hand, if the average fitness of the population remains the same or has slightly changed in comparison to previous generations, the population is slowly evolving and therefore individuals are widely spread throughout far-off regions of the search space. Although, this analysis covers several possible scenarios, it does not include all of them. For example, individuals can be loosely located over a multi-modal landscape and the average fitness could be rapidly increasing or vice versa.

Researchers have demonstrated that the exploration-exploitation trade-off can be implicitly controlled in cGAs through the configuration of the population topology and the neighbourhood [27,28]. A measure known as the neighbourhood-grid ratio (NGR) has been defined in order to describe its behaviour. The NGR was first introduced by [15]. Its calculation corresponds to measuring the dispersion of p points (each individual position) centred at (x_0,y_0). A dispersion measure is used because other possible measures, such as the radius of a circle, would give the same value for different population topologies. The dispersion of a points pattern is calculated as follows:

$$D=\sqrt{\frac{\sum(x_i-\bar{x})^2+\sum(y_i-\bar{y})^2}{p}} \tag{1}$$

where $(\bar{x})=\frac{\sum_{i=1}^{p}x_i}{p}$ and $(\bar{y})=\frac{\sum_{i=1}^{p}y_i}{p}$. Hence, the NGR is given by

$$NGR=\frac{D_{neighbourhood}}{D_{grid}} \tag{2}$$

In Figure 1, four population configurations are drawn. Internally, smaller cellular structures imply a higher or lower NGR, that as a whole would modify the search making it more exploitative or explorative. A $n\times n$ configuration, where n is the number of individuals per grid side, is the only topology which NGR directly represents its level of selective pressure. A $2(n/2\times n)$ configuration divides a square grid in two rectangular topologies with $n/2$ individuals on the shorter side. A $4(n/2\times n/2)$ configuration divides a fully connected square topology in four smaller square toroidal grid configurations with $n/2$ individuals per side. Finally, linear toroidal arrays are formed with length of $n/2$ individuals. Configurations denoted by $n\left(1\times\frac{n}{2}\right)$ and $2\left(\frac{n}{2}\times n\right)$ are implemented in both vertical and horizontal alignments. These lattice configurations have been used in the experimental set-up. In order to assess these internally different configuration topologies in terms of selective pressure, an empirical study of the take-over times is necessary.

For most topologies in Figure 1, a compact Von Neumann local neighbourhood is used consisting of individuals located at North, East, South and West with a Manhattan distance of one from the central individual; except for linear lattices ($n\left(1\times\frac{n}{2}\right)$) that employ

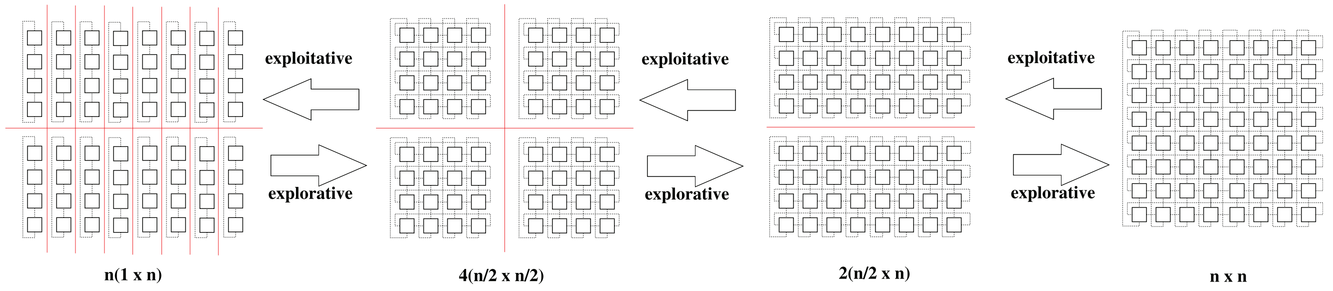


Figure 1. Neighbourhood-grid ratios corresponding to cellular structures and sub-structures. Each topology configuration induces a different level of selective pressure. Applying constant or adaptive *internal* reconfiguration criteria globally affects the search making it more explorative or exploitative while maintaining adjacency among individuals.
doi:10.1371/journal.pone.0041279.g001

a linear local neighbourhood with a Manhattan distance of two. As an example, for a population of 400 individuals, dispersion measures are $D_{vonNeumman}=0.8944$ with local Von Neumman neighbourhood and $D_{linear}=1.4142$ for the linear case. If individuals are distributed on a 20×20 square topology, corresponding NG ratios, as regards each independent cellular structure, are as follows:

- $NGR_{n\times n}=0.1093$
- $NGR_{\left(\frac{n}{2}\times n\right)}=0.1380$
- $NGR_{\left(\frac{n}{2}\times\frac{n}{2}\right)}=0.2169$
- $NGR_{\left(1\times\frac{n}{2}\right)}=0.4780$

Different ratios means having higher or lower selection pressures. Therefore, decentralized GAs are structurally capable of modifying selective pressure while changing the population's structural configuration. Lower ratios means more exploration is performed while higher ratios imply more exploitation. However, the NGR is a representation of the selective pressure in a fully connected structure or sub-structure. Therefore, it does not reflect the overall induced selective pressure in topologies configurations formed by several cellular sub-structures, like those shown in Figure 1. In order to analyse the levels of selective pressure that each topology configuration induces, an experimental assessment is carried out to calculate the take-over times following a constant *internal* lattice reconfiguration criterion.

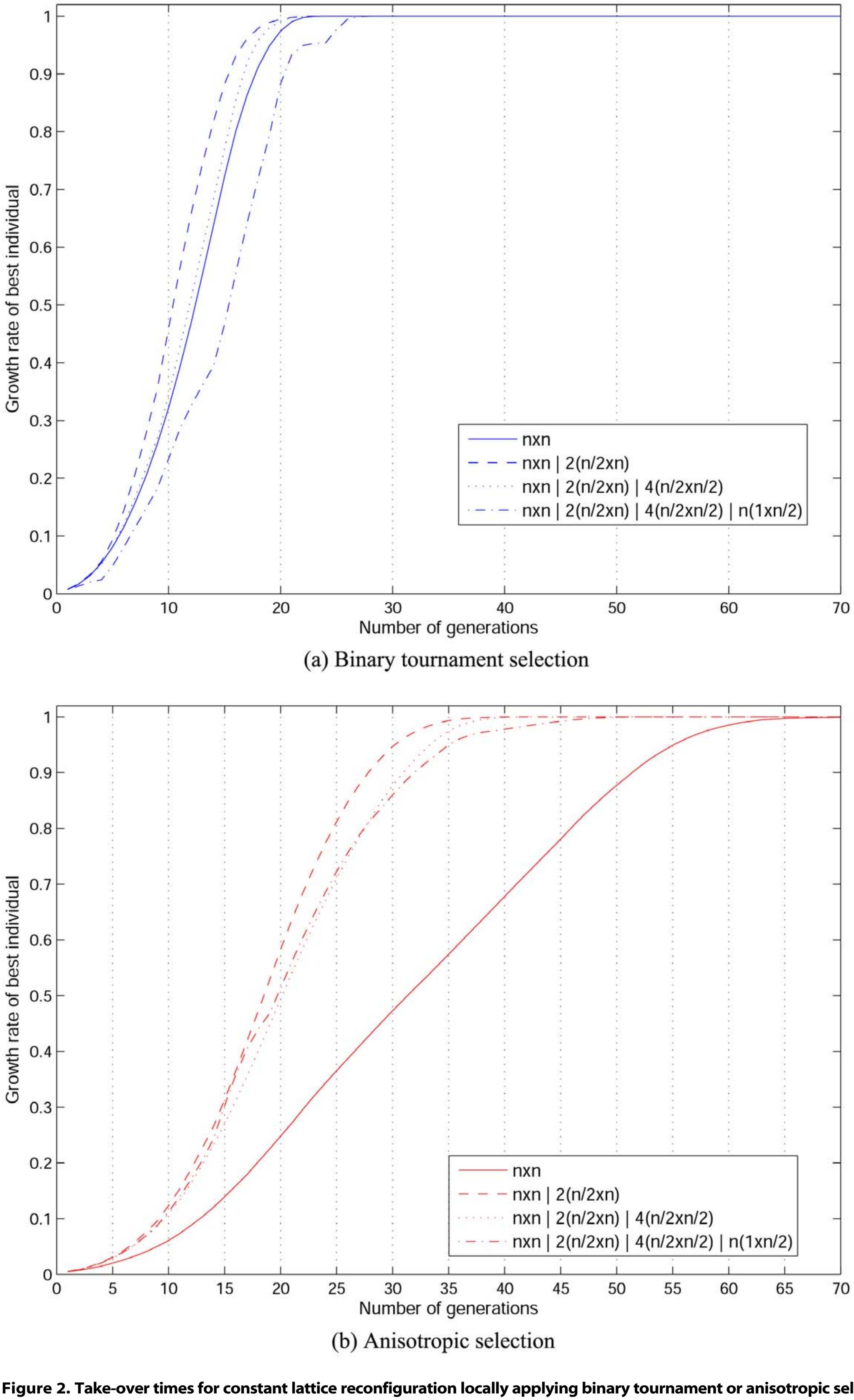
Take-over time or proportional growth of the best individual reflects how long it takes for the best individual to spread its solution throughout the whole lattice, applying only local selection. Thus, longer take-over times represent lower selection pressure and therefore more explorative behaviour. In contrast, shorter take-over times correspond to higher selection pressure, equivalent to a more exploitative search.

At a local level, two different selection methods are applied; widely known binary tournament selection and, ad-hoc for cellular GAs, anisotropic selection proposed by [29]. These selection methods provide distinctively opposite selection pressures. Binary tournament is highly exploitative in comparison to anisotropic selection which is more explorative.

In Figure 2, take-over times for both selection methods are shown. A hundred experiments with a population of 400 individuals, placed on a 20×20 grid configuration, were performed. Only constant lattice reconfiguration between a square lattice and topologies drawn in Figure 1 is performed. Constant reconfiguration occurs every 5 generations. For $n\left(1\times\frac{n}{2}\right)$ and $2\left(\frac{n}{2}\times n\right)$ formations, horizontal or vertical alignments are executed with probability $P=0.5$. Adaptive *internal* lattice reconfiguration is not subject to take-over time analysis since it depends on phenotypic or genotypic diversity; information not available when applying only local selection to a single individual. On the other hand, due to the stopping criterion which evaluates the average fitness score for the whole population, and in order to assess selective pressure on an internally different grid configuration; reconfiguration occurs between a fully connected $n\times n$ topology and one of the other configuration options.

For both local selection methods in Figure 2, solid lines correspond to selective pressure on a fully connected $n\times n$ square topology. On a square topology, binary tournament presents higher selection pressure in comparison to anisotropic selection which is more explorative. Once constant reconfiguration is applied, anisotropic selection becomes more exploitative when either internal configuration is applied. This behaviour is not repeated through binary tournament selection, presenting a slightly more explorative behaviour when reconfiguration occurs between a square and a $n\left(\frac{n}{2}\times1\right)$ topology. However, in both selection cases when reconfiguration happens between $n\times n$ and $2\left(\frac{n}{2}\times n\right)$ topologies, the highest selection pressure is induced. Specific constraints based on take-over time analysis are considered in the proposed algorithmic approach.

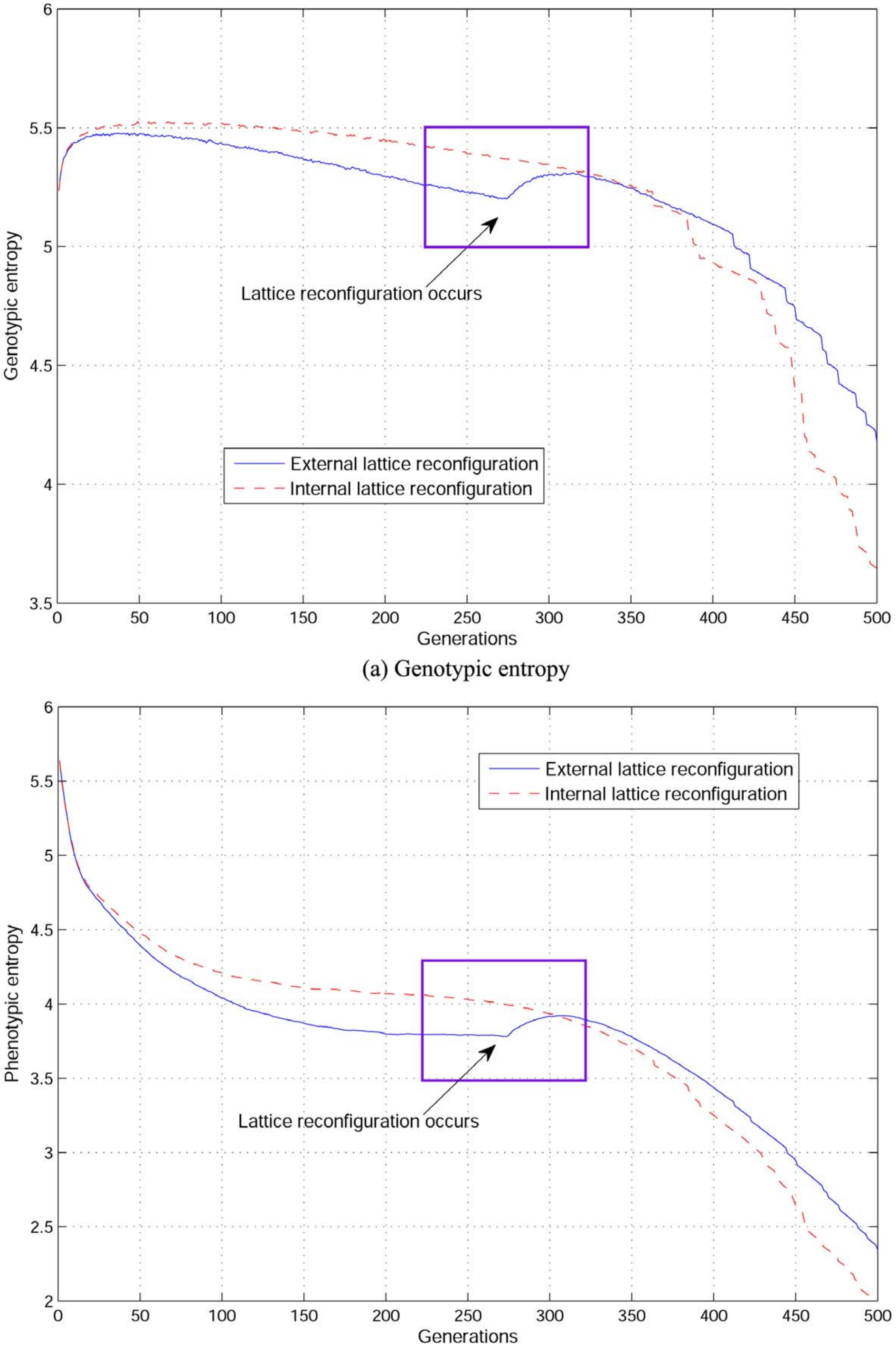
The main difference between *internal* (proposed in this article) and *external* [26] lattice reconfiguration approaches is that the former maintains individuals' adjacency while the later induces a form of explicit migration which is an influential genetic operation to restore diversity during the evolutionary process. One of the main aims in this article is to show the effect of solely manipulating the population's topology as a way to control diversity without taking advantage of other genetic operations. To show the effect of having the added effect of explicit migration, an experimental assessment on genotypic and phenotypic entropy to solve the ECC problem (see Section) is carried out. Lattice reconfiguration occurs once at a predefined time slot which is an average half the number of generations that takes for a cGA on a square topology to find



(a) Binary tournament selection

(b) Anisotropic selection

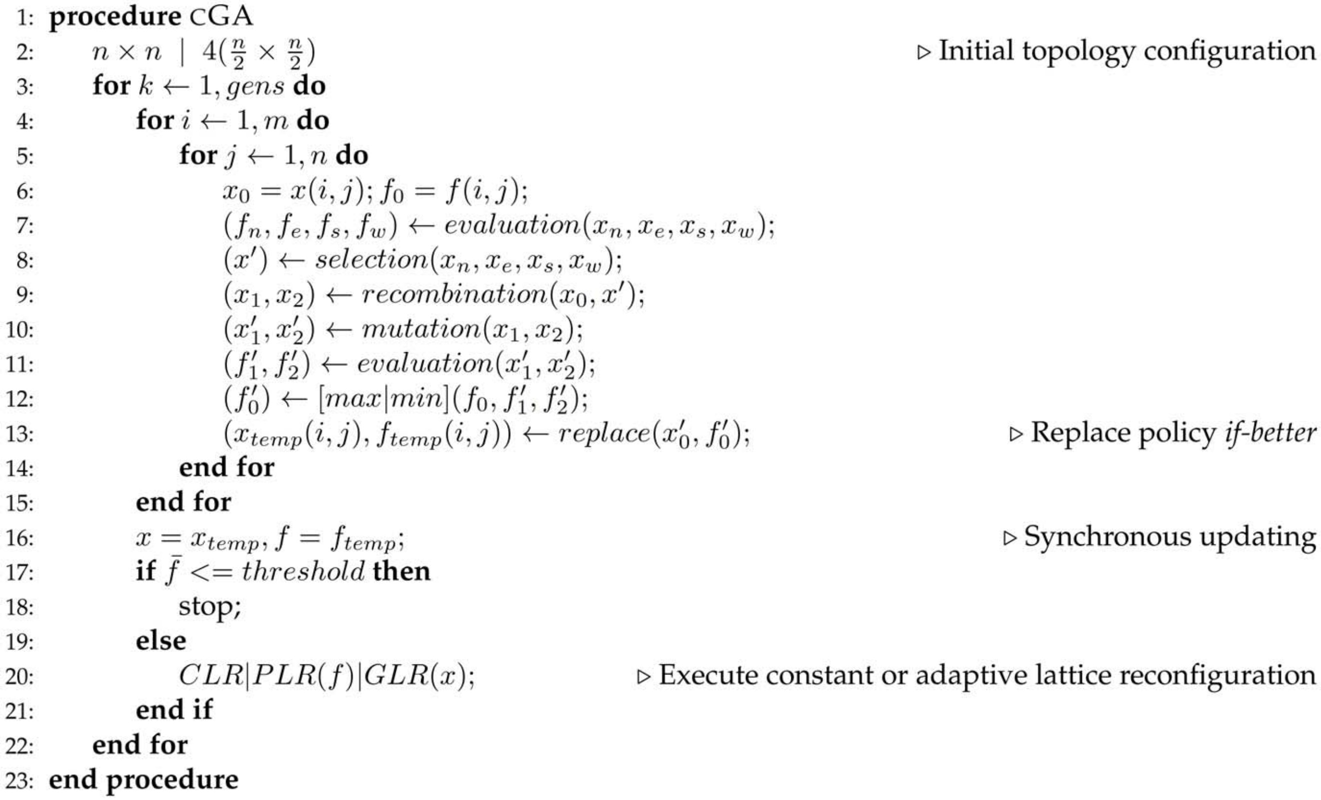
Figure 2. Take-over times for constant lattice reconfiguration locally applying binary tournament or anisotropic selection. The growth rate indicates the average number of generations for a hundred experiments that takes the best individual to spread its solution throughout the grid.
doi:10.1371/journal.pone.0041279.g002



(a) Genotypic entropy

(b) Phenotypic entropy

Figure 3. Comparing genotypic and phenotypic entropy for *internal* and *external* lattice reconfiguration.
doi:10.1371/journal.pone.0041279.g003



```
procedure cGA
    $n \times n \mid 4(\frac{n}{2} \times \frac{n}{2})$                ▷ Initial topology configuration
    for $k \leftarrow 1, gens$ do
        for $i \leftarrow 1, m$ do
            for $j \leftarrow 1, n$ do
                $x_0 = x(i,j); f_0 = f(i,j);$
                $(f_n, f_e, f_s, f_w) \leftarrow evaluation(x_n, x_e, x_s, x_w);$
                $(x') \leftarrow selection(x_n, x_e, x_s, x_w);$
                $(x_1, x_2) \leftarrow recombination(x_0, x');$
                $(x'_1, x'_2) \leftarrow mutation(x_1, x_2);$
                $(f'_1, f'_2) \leftarrow evaluation(x'_1, x'_2);$
                $(f'_0) \leftarrow [max|min](f_0, f'_1, f'_2);$
                $(x_{temp}(i,j), f_{temp}(i,j)) \leftarrow replace(x'_0, f'_0);$    ▷ Replace policy if-better
            end for
        end for
        $x = x_{temp}, f = f_{temp};$                ▷ Synchronous updating
        if $\bar{f} <= threshold$ then
            stop;
        else
            $CLR|PLR(f)|GLR(x);$        ▷ Execute constant or adaptive lattice reconfiguration
        end if
    end for
end procedure
```

Figure 4. Pseudocode for the proposed reconfigurable cellular GA.
doi:10.1371/journal.pone.0041279.g004

the problem's solution. Both reconfiguration mechanisms are assessed locally applying anisotropic selection which showed a wider span of selective pressure in contrast to binary tournament selection. For *external* lattice reconfiguration, evolution starts on a square topology and continue on a rectangular grid, making the search going from more exploitation to more exploration. On the other hand, for *internal* reconfiguration, initially the population evolves on an *internal* arrangement of four square cellular substructures ($4\left(\frac{n}{2} \times \frac{n}{2}\right)$) and lattice is then reconfigured to a square topology. For the *internal* mechanism, this means going from a more exploitative to a more explorative behaviour as shown by the take-over time experimental analysis when constant lattice reconfiguration takes place.

Entropy results are obtained from a hundred experimental samples, see Figure 3. Dashed lines correspond to *internal* reconfiguration and solid lines to *external* reconfiguration. A square shows the moment both *internal* and *external* reconfiguration is performed. For both entropy measures, change in diversity is clear when *external* reconfiguration occurs; as a result of grid change and individuals' migration. In contrast, after reconfiguration is internally performed, entropy measures do not change abruptly as a consequence of maintaining individuals' adjacency.

Another observation from entropy results in Figure 3 is that entropy measures are slightly higher for *internal* lattice reconfiguration during the initial stages of evolution and until reconfiguration occurs. However, in this experimental assessment reconfiguration is applied only once while the algorithmic approach and the experimental set-up are based on the take-over times analysis which consists in systematic reconfiguration among all proposed internal topology configurations. In the following section, details of the algorithmic procedure and diversity measures for lattice reconfiguration are provided.

Cellular Configuration

This study proposes to constantly or adaptively reconfigure the internal population topology to modify the induced selective pressure, and thus achieve an appropriate balance between searching exploration and exploitation, essential for maintaining population's diversity during evolution. In this proposal, lattice reconfiguration is implemented by subdividing a squared population into smaller squared, rectangular or linear toroidal arrays. A similar approach has been referred to as a parallel cellular GA [21]. However, that approach has a fixed *internal* configuration. Whilst an homogeneous distributed or an island model cGA is one possible perspective [30], migration policies have not been investigated nor implemented. During constant or adaptive periods of evolution, smaller topologies maintain internally toroidal connections [31,32].

In Figure 4, a pseudocode for the proposed cellular GA approach is included. A single random seed is used to generate the entire initial population, two topology configurations are set to evolve the first generations: a $n \times n$ or a $4\left(\frac{2}{n} \times \frac{2}{n}\right)$ lattice. For each individual, neighbours located at North, East, South and West positions are evaluated in order to select a second parent for reproduction. Two local selection methods, presenting opposite selection pressures, are applied for experimental purposes. Binary tournament presents higher selection pressure than anisotropic selection.

Binary tournament randomly chooses two individuals from the local neighbourhood and the best one (x') is mated with the central

```
procedure PLR,GLR($f, x$)
    if $\left|\Delta H_{P_t} - \Delta H_{P_{t-1}}\right| < \left|\Delta H_{P_{t-1}} - \Delta H_{P_{t-2}}\right|$ then    ▷ $H_G$ for genotypic entropy
        //Promote exploration
        if lattConf = $2\left(\frac{n}{2} \times n\right)$ then    ▷ Highest selective pressure
            $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$;
        else if $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$ then
            $lattConf = (n \times n)$;
        else if $lattConf = (n \times n)$ then
            $lattConf = n\left(1 \times \frac{n}{2}\right)$;
        else
            $lattConf = n\left(1 \times \frac{n}{2}\right)$;    ▷ Lattice configuration remains the same
        end if
    else if $\left|\Delta H_{P_t} - \Delta H_{P_{t-1}}\right| > \left|\Delta H_{P_{t-1}} - \Delta H_{P_{t-2}}\right|$ then
        //Promote exploitation
        if $lattConf = n\left(1 \times \frac{n}{2}\right)$ then    ▷ Lowest selective pressure
            $lattConf = (n \times n)$
        else if $lattConf = (n \times n)$ then
            $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$;
        else if $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$ then
            $lattConf = 2\left(\frac{n}{2} \times n\right)$;
        else
            $lattConf = 2\left(\frac{n}{2} \times n\right)$;    ▷ Lattice configuration remains the same
        end if
    else
        ▷ 0.5 probability to reconfigure to a more explorative or exploitative topology
    end if
end procedure
```

Figure 5. Pseudocode for phenotypic or genotypic lattice reconfiguration through binary tournament selection.
doi:10.1371/journal.pone.0041279.g005

individual (x_0). On the other hand, anisotropic selection requires an extra parameter (α) that leads the search in North-South or East-West grid directions. Thus, anisotropic probabilistic equations are defined for each individual in the neighbourhood as: $P_n = P_s = P_0(1+\alpha)$ and $P_e = P_w = P_0(1-\alpha)$; where P_0 is the uniform probability for each neighbour to be selected. If $\alpha = 0.8$ and $P_0 = 0.25$, individuals at North and South positions are assigned with higher probabilities for selection compared to East and West individuals that remain with lower selection probabilities. In this way the search process is directionally guided and adjusting the α parameter would supply higher or lower selection pressure. For experimental purposes, $\alpha = 0.8$ has been used due to presenting lower selection pressure and thus being more explorative. Moreover, if $P_0 = 0.0$ all neighbours have same selection probability and anisotropic selection behaves as binary tournament selection. Although fewer studies have been reported using this local selection method in cGAs, anisotropic selection does provide more flexibility in terms of selective pressure, making use of cGAs' structural properties, and even more when the population topology is reconfigured; this being the main perspective in this research [29,33].

Recombination is performed using Single Point Crossover (SPC) with highest probability and constant mutation probability. Both genetic operators have been applied in their simplest form, as it is not the objective of this research to evaluate the effect of either of them but to widely investigate the effect of dynamically changing selection pressure through cGAs structural properties. Although it is outwith the scope of this paper, particular attention should be paid to mutation as its effect has been modelled for cellular GAs as an essential control parameter to lead or mislead the search [34,35]. Replacement takes place following an *if-better* policy. Only if.

In Figure 4, line 20, $[CLR|PLR(f)|GLR(x)]$ corresponds to constant lattice reconfiguration, phenotypic lattice reconfiguration or genotypic lattice reconfiguration, respectively.

CLR consists in constant cyclic reconfiguration among internal lattice topologies shown in Figure 1. Every c generations the topology is reconfigured to a different topology configuration following a cyclic pattern. For rectangular ($2\left(\frac{n}{2} \times n\right)$) and linear ($n\left(1 \times \frac{n}{2}\right)$) grid configurations, both horizontal and vertical alignments are performed with 50% probability. The overall induced selective pressure varies according to the take-over time analysis presented in previous section.

Grid reconfiguration based on phenotypic diversity (PLR procedure) evaluates differences in phenotypic entropy, H_P, among consecutive generations:

$$H_P = -\sum_{j=1}^{N} f_j \log(f_j) \tag{3}$$

```
procedure PLR,GLR($f, x$)
    if $\left|\Delta H_{P_t} - \Delta H_{P_{t-1}}\right| < \left|\Delta H_{P_{t-1}} - \Delta H_{P_{t-2}}\right|$ then        ▷ $H_G$ for genotypic entropy
        //Promote exploration
        if lattConf = $2\left(\frac{n}{2} \times n\right)$ then        ▷ Highest selective pressure
            $lattConf = n\left(1 \times \frac{n}{2}\right)$;
        else if $lattConf = n\left(1 \times \frac{n}{2}\right)$ then
            $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$;
        else if $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$ then
            $lattConf = (n \times n)$;
        else
            $lattConf = (n \times n)$;        ▷ Lattice configuration remains the same
        end if
    else if $\left|\Delta H_{P_t} - \Delta H_{P_{t-1}}\right| > \left|\Delta H_{P_{t-1}} - \Delta H_{P_{t-2}}\right|$ then
        //Promote exploitation
        if $lattConf = (n \times n)$ then        ▷ Lowest selective pressure
            $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$;
        else if $lattConf = 4\left(\frac{n}{2} \times \frac{n}{2}\right)$ then
            $lattConf = n\left(1 \times \frac{n}{2}\right)$;
        else if $lattConf = n\left(1 \times \frac{n}{2}\right)$ then
            $lattConf = 2\left(\frac{n}{2} \times n\right)$;
        else
            $lattConf = 2\left(\frac{n}{2} \times n\right)$;        ▷ Lattice configuration remains the same
        end if
    else
                ▷ 0.5 probability to reconfigure to a more explorative or exploitative topology
    end if
end procedure
```

Figure 6. Pseudocode for phenotypic or genotypic lattice reconfiguration through anisotropic selection.
doi:10.1371/journal.pone.0041279.g006

where f_j is the proportion of individuals in one generation having fitness j. Thus, the difference between ΔH_{P_t}, $\Delta H_{P_{t-1}}$ and $\Delta H_{P_{t-2}}$ would determine if phenotypic diversity in current generation has or has not significantly changed with respect to previous generations; where $\Delta H_{P_t} = |H_{P_t} - H_{P_{t-1}}|$.

In Figures 5 and 6, adaptive reconfiguration mechanisms for binary tournament and anisotropic local selection methods are described. If diversity among phenotypes decreases, $|\Delta H_{P_t} - \Delta H_{P_{t-1}}|$ is less than $|\Delta H_{P_{t-1}} - \Delta H_{P_{t-2}}|$ then exploration should be encouraged by switching to a lattice configuration that presents in average lower selective pressure. If the same condition indicates an increase on phenotypic diversity, in order to make the search more aggressive, exploitation should be promoted by increasing selection pressure. Otherwise, population's topology remains the same.

On the other hand, determining genotypic diversity requires to calculate the Hamming distance among chromosomes. The distance between current ($\Delta\bar{H}_{G_t} = |\bar{H}_{G_t} - \bar{H}_{G_{t-1}}|$) and previous ($\Delta\bar{H}_{G_{t-1}}$, $\Delta\bar{H}_{G_{t-2}}$) generations is calculated in order to assess genotypic diversity changes. In a similar way to phenotypic diversity, if $|\Delta\bar{H}_{G_t} - \Delta\bar{H}_{G_{t-1}}|$ difference is less than $|\Delta\bar{H}_{G_{t-1}} - \Delta\bar{H}_{G_{t-2}}|$, exploration should be encouraged through a lattice configuration which induces a lower selective pressure. On contrast, exploitation should be promoted by increasing overall selection pressure.

For both diversity metrics, if the change rate of phenotypic or genotypic diversity equals zero, it means the search process has reached a plateau and no more competitive solutions would arise. Therefore, the effect of reconfiguring the grid will not significantly affect the evolutionary process. Because the proposed adaptive criteria do not evaluate the growing or decaying direction of diversity, the algorithmic approach follows a 0.5 probability policy to reconfigure the grid to an internal configuration that provides either a more exploitative or more explorative behaviour.

In [26], conditions for lattice *external* reconfiguration imply the relocation of individuals to new positions, inducing a kind of migration among them. Hence, the impact of reconfiguring the grid is not completely clear; as migration is an influential operation that can by itself change significantly the performance of cGAs [18,19]. Reconfiguration mechanisms proposed here, do not induce any explicit migration and once topology configuration has *internally* reconfigured, a toroidal connection with wraparound edges is maintained in all cellular sub-structures. Thus, higher or lower intensities of selective pressure are applied throughout the landscape after lattice reconfiguration takes place while individuals' adjacency is maintained.

Pre-programmed criterion by Dorronsoro et. al. consists in *externally* changing topology's shape after a certain number of

Table 1. Benchmark problems.

Problem	Fitness function	Properties
Rastrigin	$f(\vec{x})=10q+\sum_{i=1}^{q}\left(x^2-cos(2\pi x_i)\right)$	$q=10$, $\bar{f}_T\leq 0.0005$, multi-modal, regular
Griewank	$f(\vec{x})=1+\sum_{i=1}^{q}\frac{x_i^2}{4000}-\Pi\, cos\left(\frac{x_i}{\sqrt{i}}\right)$	$q=10$, $\bar{f}_T\leq 0.0001$, multi-modal, epistatic, regular
Langerman	$f(\vec{x})=-\sum_{i=1}^{q}c_i e^{-\frac{1}{\pi}\sum_{j=1}^{D}\left(x_j-a_{ij}^2\right)}cos\left(\pi\sum_{j=1}^{D}\left(x_j-a_{ij}\right)^2\right)$	$q=10$, $\bar{f}_T\geq -1.4990$, multimodal, epistatic, non-regular
FMS	$f(t)=a_1\cdot sin(w_1\cdot t\cdot\theta+a_2\cdot sin(w_2\cdot t\cdot\theta+a_3\cdot sin(w_3\cdot t\cdot\theta)))$	$q=6$, $\bar{f}_T\leq 0.04$, multimodal, epistatic, non-regular
SLE	$A\vec{x}=b$	$q=10$, $\bar{f}_T\leq 0.04$, epistatic
GPS	$f(\vartheta,\beta,b)=\sum_{i=1}^{m}\sum_{j=2}^{n}\cos\left(\frac{2\pi}{\lambda}\left(\frac{DD\Phi_{AB}^{1j}}{m(n-1)}-\frac{DD\Phi_{AB}^{1j}(\varphi,\beta,b)}{m(n-1)}\right)\right)$	$q=3$, $\bar{f}_T\geq 0.99$, multimodal, epistatic, non-regular
MMDP	$f_{MDDP}(\vec{x})=\sum_{i=1}^{q}fitness_{x_i}$	$q=25$, q sub-problems, $\bar{f}_T\geq 0.99$, multi-modal, combinatorial
MTTP	$f_{MTTP}(\vec{x})=\sum_{i=1}^{p}weight_{x_i}$	$q=100$, $\bar{f}_T\leq 0.005$, epistatic, NP-combinatorial
P-Peaks	$f_{P-Peaks}(\vec{x})=\frac{1}{Q}max_{i=1}^{P}(Q-Hamming(\vec{x},P_i))$	$P=100$ peaks, $Q=100$ bits, $\bar{f}_T=1.0$, non-regular, non-manually tunable
ECC	$f_{ECC}(\vec{C})=\frac{1}{\sum_{i=1}^{M}\sum_{j=1;i\neq j}^{M}\frac{1}{d_{ij}^2}}$	$M=24$ codewords, $\bar{f}_T=0.0674$, non-regular, non-manually tunable

doi:10.1371/journal.pone.0041279.t001

generations, that is half way through a successful run for a square topology. For the adaptive approach, authors determined a constant parameter as a threshold to measure phenotypic and genotypic diversity and switch among topology configurations [26]. Criteria herein proposed avoid the use of an extra parameter and directly detect significant changes while extending the generation-range to measure diversity. In the following section, results analysis including the experimental set-up is presented.

Table 2. Convergence time and hit rate results for continuous, FMS and SLE problems through static, preprogrammed (*external*) and constant (*internal*) lattice reconfiguration with *binary tournament local selection.*

Static/Preprog.					
***External* Reconfig.**	**Rastrigin**	**Griewank**	**Langerman**	**FMS**	**SLE**
Square	136.68±9.48	272.79±46.45	291.04±103.92	210.78±58.76	297.28±72.56
	100%	89%, 3.12	21%, 4.07	41%, 4.91	39%, 4.87
Rectangular	148.01±11.49	315.41±63.12	285.52±98.34	247.70±80.70	359.09±87.22
	100%	90%, 3	19%, 3.92	48%, 4.99	31%, 4.62
Narrow					
219.12±22.56	412.74±45.09	417.33±71.00	406.52±64.75	573.25±67.44	
	100%	63%, 4.82	3%, 1.70	34%, 4.73	16%, 3.66
Square-Narrow					
	150.23±21.96	313.48±83.23	226.31±76.08	206.18±46.72	347.55±119.30
	37%, 4.82	100%	19%, 4.20	19%, 3.92	29%, 4.53
Narrow-Square	139.26±13.09	262.08±34.10	305.35±100.08	207.42±40.20	298.66±89.93
	100%	96%, 1.95	17%, 3.75	35%, 4.76	27%, 4.43
Constant					
***Internal* Reconfig.**					
Square→Constant	122.79±8.33	245.35±43.90	318.73±164.18	195.23±54.12	257.09±77.29
	100%	80%, 4.00	23%, 4.20	43%, 4.95	21%, 4.07
$4(\frac{n}{2}\times\frac{n}{2})$→Constant	123.09±9.84	245.86±38.15	222.90±88.46	189.78±46.32	269.75±79.87
	100%	73%, 4.43	20%, 4.00	42%, 4.93	28%, 4.49
[3]ANOVA/K-W	+	+	•	•	+

doi:10.1371/journal.pone.0041279.t002

Table 3. Convergence time and hit rate results for continuous, FMS and SLE problems through *external* and *internal* adaptive lattice reconfiguration with *binary tournament local selection.*

Adaptive Phenotypic					
***External* Reconfig.**	**Rastrigin**	**Griewank**	**Langerman**	**FMS**	**SLE**
Rectangular→Adapt.					
	135.31 ± 8.45	263.63 ± 39.72	261.27 ± 106.77	178.60 ± 29.73	304.72 ± 83.51
	100%	88%, 3.24	29%, 4.53	53%, 4.99	29%, 4.53
Narrow→Adapt.	137.00 ± 10.20	260.94 ± 43.13	218.20 ± 62.26	185.44 ± 35.87	298.76 ± 104.25
	100%	84%, 3.66	24%, 4.27	47%, 4.99	26%, 4.33
Internal Reconfig.					
Square→Adapt.	127.33 ± 10.6	254.56 ± 42.65	234.50 ± 113.11	184.97 ± 49.29	293.37 ± 89.32
	100%	74%,4.38	18%,3.84	47%,4.99	35%,4.76
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	121.88 ± 10.09	257.52 ± 43.61	231.44 ± 93.80	165.00 ± 32.65	291.09 ± 112.90
	100%	70%, 4.58	18%, 3.84	52%, 4.99	32%, 4.66
Adaptive Genotypic					
***External* Reconfig.**					
Rectangular→Adapt.	126.66 ± 8.33	250.05 ± 39.85	276.94 ± 111.51	167.39 ± 26.65	232.50 ± 70.52
	100%	76%, 4.27	19%, 3.92	46%, 4.98	26%, 4.38
Narrow→Adapt.	125.77 ± 8.42	253.73 ± 38.81	237.55 ± 72.31	179.95 ± 33.41	274.73 ± 90.69
	100%	76%, 4.27	20%, 4.00	49%, 4.99	23%, 4.20
Internal Reconfig.					
Square→Adapt.	125.17 ± 10.39	234.54 ± 38.60	221.90 ± 67.21	199.97 ± 53.37	270.83 ± 100.29
	100%	75%, 4.33	21%, 4.07	37%, 4.82	30%, 4.58
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	127.26 ± 11.62	256.50 ± 39.57	259.47 ± 114.53	185.18 ± 44.79	283.94 ± 80.38
	100%	78%, 4.14	17%, 3.75	37%, 4.82	35%, 4.76
[3]ANOVA/K-W	+	+	•	•	•

doi:10.1371/journal.pone.0041279.t003

Results

Experimental Set-up

In order to assess the dynamic reconfiguration criteria proposed in this article and as an initial comparison, pre-programmed and adaptive criteria of Dorronsoro et al. have been applied to the benchmark problems tackled in this research [26].

A comparison between centralized versus decentralized GAs is outwith the scope of this article. However, several authors have demonstrated in previous research that cellular GAs outperform serial GAs when solving difficult optimization tasks [8,12,21]. In this article, as a reference point, a traditional toroidal square topology is used for performance comparison.

Benchmark Problems

In Table 1, a summary of mathematical equations and properties of the benchmark problems is included. Three continuous problems: the Rastrigin, the Griewank and the Langerman functions are assessed [36,37]. Three real-world problems are tackled: the Frequency Modulation Sound (FMS) problem, the System of Linear Equations (SLE) problem [21] and the GPS attitude determination problem which has been previously tackled by the authors in [38]. Finally, three combinatorial problems are evaluated: the Massively Multi-modal Deceptive Problem (MMDP), the Minimum Tardy Task Problem (MTTP), the P-Peaks problem [39] and the Error Correcting Code (ECC) design problem [21]. Chromosome encoding for all problems is binary. The stop condition is defined by the average population's fitness ($\bar{f}_T$) score for each problem.

Experimental Constraints

Crossover and mutation operators are configured in a simple form and are kept constant during the search process. Single Point Crossover (SPC) operates pairs formed by a central individual and an individual selected from the local neighbourhood with probability $P_c = 1.0$. SPC randomly selects a single point for recombination. On the other hand, mutation is performed after recombination with probability $P_m = 0.02$. Neither genetic operators are specifically evaluated in this article and are configured in their simplest form.

Replacement policies also play an important role in GAs. In cGAs, an offspring can replace a current individual in the following three ways: 1) always, 2) only if it has a better fitness score or 3) following some other condition. In this research an *only replace if better* policy is followed. On the other hand, synchronous population updating is also considered. Synchronous updating requires current and next generations to be stored separately; once all individuals have evolved, new offspring will entirely replace the current population. Asynchronous updating has also been investigated and in several problems outperforms synchronous updating in terms of convergence time although the accuracy of the results and their hit rate are negatively affected [20].

Table 4. Convergence time and hit rate results for GPS and combinatorial problems through static, preprogrammed (*external*) and constant (*internal*) lattice reconfiguration with *binary tournament local selection*.

Static/Preprog.					
***External* Reconfig.**	**GPS**	**MMDP**	**MTTP**	**P-Peaks**	**ECC**
Square	73.12±28.69	431.31±38.45	259.65±44.16	169.46±19.99	1522.56±396.86
	57%, 4.95	89%, 3.12	99%, 0.99	100%,	16%, 3.66
Rectangular	85.24±30.04	517.14±63.87	295.57±48.70	175.52±23.11	1127.4±683.66
	53%, 4.99	90%, 3.00	99%, 0.99	100%,	30%, 4.58
Narrow	123.57±43.77	696.00±0.00	− − − −	170.34±23.36	972.83±339.27
	66%, 4.73	1%, 0.99	− −	100%,	6%, 2.37
Square-Narrow	98.84±45.76	455.56±84.93	314.85±69.92	209.58±30.71	1330.1±574.11
	46%, 4.98	86%, 3.46	97%, 1.70	100%,	29% 4.53
Narrow-Square	85.62±32.19	488.57±51.19	276.41±48.50	242.25±44.63	1605.23±468.63
	58%, 4.93	92%, 2.71	99%, 0.99	100%,	17%, 3.75
Constant					
***Internal* Reconfig.**					
Square→Constant	72.55±30.18	358.03±41.76	243.08±35.77	170.68±21.00	849.10±373.14
	54%, 4.98	88%, 3.24	98%, 1.70	100%,	76%, 4.27
$4(\frac{n}{2}\times\frac{n}{2})\rightarrow$Constant	71.00±20.22	352.72±54.93	252.16±43.58	173.99±25.04	942.38±440.91
	53%, 4.99	87%, 3.36	100%,	100%,	86%, 3.46
[3]ANOVA/K-W	•	+	+	•	+

doi:10.1371/journal.pone.0041279.t004

The following experimental constraints are evaluated:

- A population size of 400 individuals was used for most problems. Due to their size, GPS and MTTP problems are tackled using a population size of 64 and 100 individuals respectively.
- Binary encoding is used in all problems. In continuous and real problems, each variable is encoded in 10 bits, except for the GPS attitude problem where each variable is encoded in 14, 10, 8 bits respectively. Binary encoding is implicit in combinatorial problems.
- Local neighbourhood configuration is Von Neumann or linear composed by four individuals plus a central individual.
- One hundred independent runs are carried out per experimental reconfiguration criterion.
- A problem specific threshold based on population average fitness is used as a stop condition.
- A limit of 500 generations is used in most problems, except for the Langerman function, the SLE and the MMDP problems with a limit of 700 and the ECC problem with 2500 generations.

The same reconfiguration mechanisms are applied to all problems and results are analysed based on convergence time and hit rate. The accuracy of results is determined by problem specific thresholds, see Table 1. The stop condition evaluates population's average fitness score. An accepted stop condition considers that experimental samples should evolve until a solution of the same quality is found and not after a certain number of generations [20]. The stop condition herein considered has been defined due to previous research constraints related to fault tolerance. Although fault tolerance is a different topic, outwith the scope of this article, the decision was made to keep the same stop condition for consistency. Results herein obtained will be taken up again for future research in the area of fault tolerance. For further reference readers can consult [38,40].

Statistical Tests

In order to statistically support results so far obtained, the Lilliefors normality test, at 5% of significance, is performed on each set of convergence time results. This test is suitable when a fully-specified null distribution is unknown, in contrast to the Kolmorov-Smirnov test. Once normality of results has been established, an Analysis of Variance (ANOVA) is applied among results that follow a normal distribution whereas the Kruskal-Wallis test is applied to results that are not normally distributed. For sets of results with non-normal distributions, dispersion of data is calculated using the *mean absolute deviation* and *italics* are used to highlight these cases in the results tables in the next section.

In the next section, results are summarized in two tables, one for static, pre-programmed and constant reconfiguration criteria and a second one for adaptive approaches. Then, in order to prove that a certain approach outperforms others, in terms of convergence time, a multiple comparison test is carried out, after a statistical analysis is done independently for each table of results. In contrast, if statistical proof is obtained only in one of the two tables, a multiple comparison test is applied separately.

In the results tables, where a statistically significant difference at 5% has been found among convergence time results, this has been represented by the symbol +. On the contrary, a • symbol represents results which are not statistically different, and where therefore the application of proposed dynamic criteria makes no difference in terms of the number of generations. On the other hand, having statistically different results does not mean the proposed approaches definitively improve cGAs performance.

Table 5. Convergence time and hit rate results for GPS and combinatorial problems through *external* and *internal* adaptive lattice reconfiguration with *binary tournament local selection*.

Adaptive Phenotypic					
***External* Reconfig.**	**GPS**	**MMDP**	**MTTP**	**P-Peaks**	**ECC**
Rectangular→Adapt.	68.78±29.60	411.24±42.25	231.58±34.06	180.50±24.84	976.75±398.04
	60%, 4.89	90%, 3.00	100%,	100%,	83%, 3.75
Narrow→Adapt.	65.58±20.44	427.91±57.66	226.52±36.68	181.51±25.91	932.58±448.57
	53%, 4.99	89%, 3.12	100%,	100%,	87%, 3.36
Internal Reconfig.					
Square→Adapt.	68.96±21.90	377.21±51.26	240.07±33.26	178.69±28.34	1086.05±511.28
	50%, 5.0	85%, 3.57	100%,	100%,	78%,
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	73.05±31.85	373.81±42.29	239.09±37.06	167.06±22.38	925.50±410.92
	51%, 4.94	85%, 3.57	99%, 0.99	100%,	84%, 3.66
Adaptive Genotypic					
***External* Reconfig.**					
Rectangular→Adapt.	63.00±22.51	366.14±43.70	227.22±32.77	177.55±23.93	858.18±413.24
	58%, 4.93	91%, 2.86	100%,	100%,	91%, 2.86
Narrow→Adapt.	73.12±31.02	360.77±40.39	222.55±31.01	166.65±22.66	949.74±433.50
	56%, 4.96	94%, 2.37	99%, 0.99	100%	89%, 3.12
Internal Reconfig.					
Square→Adapt.	74.73±33.31	374.96±43.94	241.17±37.42	173.70±22.25	1098.69±575.46
	57%, 4.95	88%, 3.24	100%,	100%,	84%, 3.66
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	64.35±18.60	373.49±50.99	247.12±36.14	175.32±25.55	1006.97±443.49
	59%, 4.91	87%, 3.36	100%,	100%,	84%, 3.66
[3]ANOVA/K-W	•	+	+	•	+

doi:10.1371/journal.pone.0041279.t005

This statistical analysis only evaluates the efficiency of proposed techniques. Thus, the efficacy of results has also been statistically considered.

The hit rate has been statistically evaluated as a Bernoulli trial. A random experiment whose result is either success or failure can be considered a Bernoulli trial [12]. For each dynamic lattice reconfiguration criterion, 100 experiments were carried out. In successful experiments, the global optima, at a certain threshold, is reached. Measuring the standard deviation of successful experiments percentages provides a numerical value that indicates how significantly different are these success search rates. The standard deviation for each experimental sample is calculated as follows:

$$\sigma=\sqrt{r*\hat{p}(1-\hat{p})} \tag{4}$$

where $\hat{p}$ represents the probability of successful experiments and r is the total number of experiments. In results tables, the standard deviation is shown next to hit rates percentages for each experimental sample. In the next section, results are detailed.

Results Analysis

Results analysis aims to empirically support the main hypothesis of this study: dynamic modification of the internal configuration of the population's topology would lead to a significant improvement in cGAs' performance. At a local level, as explained previously in Section, two different local selection methods have been applied. The main difference between them consists of inducing higher (binary tournament) or lower (anisotropic) selection pressure. These two methods were chosen because of their considerably different selection intensity. Thus, it is possible to separately evaluate 1) the effect of having a highly exploitative or explorative cellular GA, due to local selection, and 2) the added flexibility of dynamically reconfiguring the population topology in order to supply the search process with a more convenient and balanced exploitation-exploration trade-off.

For presentation, results are divided by local selection method. Two tables of results are organized: the first includes static, pre-programmed (*external*) reconfiguration and constant (*internal*) reconfiguration. The second group presents adaptive approaches for both *external* and *internal* lattice reconfiguration approaches.

In the tables, numbers in **bold** indicate the best average convergence time and *italics* are used to distinguish between standard deviation (normal distributions) and mean absolute deviation (non normal distributions) as dispersion measures. Next to the hit rates, corresponding Bernoulli trial standard deviations are shown in small numbers.

Binary Tournament Local Selection

Tables 2 and 3 present results for static, pre-programmed, constant and adaptive lattice reconfiguration criteria through binary local selection. The Rastrigin function hit rate is 100% for all configurations, and convergence time is slightly affected through lattice reconfiguration. Statistical proof is obtained for the two best criteria: square → constant and $4\left(\frac{n}{2}\times\frac{n}{2}\right)$ →

Table 6. Convergence time and hit rate results for continuous, FMS and SLE problems through static, preprogrammed (*external*) and constant (*internal*) lattice reconfiguration with *anisotropic local selection*.

Static/Preprog.					
***External* Reconfig.**	**Rastrigin**	**Griewank**	**Langerman**	**FMS**	**SLE**
Square	230.38±15.51	438.36±39.04	432.61±47.01	379.38±58.58	594.21±54.32
	100%	68%, 4.66	21%, 4.07	52%, 4.99	19%, 3.92
Rectangular	283.34±27.42	464.18±23.06	429.35±61.08	603.66±54.68	
	100%	16%, 3.66	1%, 0.99	14%, 3.46	6%, 2.37
Narrow	470.77±19.00	– – – –	– – – –	– – – –	– – – –
	53%	– –	– –	– –	– –
Square-Narrow	279.35±29.43	437.57±52.99	430.50±34.54	368.39±75.91	585.87±115.39
	100%	14%, 3.46	6%, 2.37	33%, 4.70	8%, 2.71
Narrow-Square	246.43±14.76	431.53±36.83	448.11±36.08	420.51±53.73	600.17±79.09
	100%	93%, 2.55	18%, 3.84	35%, 4.76	29%, 4.53
Constant					
***Internal* Reconfig.**					
Square→Constant	186.89±14.56	370.11±45.66	316.62±66.14	305.24±61.08	469.50±114.32
	100%	89%, 3.12	37%, 4.82	66%, 4.73	42%, 4.93
$4(\frac{n}{2} \times \frac{n}{2})$→Constant	186.67±11.48	375.97±45.76	309.42±69.13	300.32±53.09	442.80±93.42
	100%	95%, 2.17	26%, 4.38	71%, 4.53	42%, 4.93
^{3}ANOVA/K-W	+	+	+	+	+

doi:10.1371/journal.pone.0041279.t006

adaptive phenotypic *internal* reconfiguration. On the other hand, adaptive approaches do not show significant difference in convergence time.

Similar results in the number of generations are obtained for the Griewank function, statistical difference at 5% is proved for square → constant *internal* reconfiguration with respect to static and preprogrammed criteria. However, in terms of success search rate, pre-programmed square → narrow approach shows major advantage in this regard, from 80% ($\sigma = 4.00$) to 100% $\sigma = 0.0$, based on calculated sample standard deviations. On the other hand, for adaptive approaches the best convergence time is obtained by square → adaptive genotypic *internal* reconfiguration with statistical difference. The best hit rate is 88% ($\sigma = 3.24$) for rectangular → adaptive *external* reconfiguration, which is not as good as the pre-programmed approach.

The Langerman function is the most difficult in the continuous set of problems tackled in this study; hit rates are not higher than 29%. The improvement in results for lattice reconfiguration approaches is not significant, either in convergence time or in the hit rate for all proposed criteria through binary tournament local selection. Similar results were obtained for the FMS problem. No significant improvement in terms of convergence time was obtained through any of the proposed approaches. Standard deviations for hit rates do not show significant statistical difference either.

From these two sets of results, it is worth noticing that the best convergence times in Table 2 correspond to constant approaches. However, hit rates in some cases are higher for either static or pre-programmed criteria, with statistical proof, for example, in the Griewank function. For adaptive approaches, in Table 3, best convergence times correspond to *external* reconfiguration techniques, however these results are not statistically supported. Results consistency for convergence times is maintained for most problems, except in the Langerman function. In terms of convergence time, the SLE problem presents a significant difference for square → constant *internal* reconfiguration. However, in terms of hit rate, traditional static square grid provides the highest efficacy (39%). In contrast, the best adaptive square → adaptive and $4(\frac{n}{2} \times \frac{n}{2})$ → adaptive *internal* phenotypic and genotypic based criteria reach a hit rate of 35%, with no meaningful difference due to $\sigma = 4.87$ and $\sigma = 4.76$ respectively. The convergence times for adaptive criteria are not different, and the consistency of the results is not well maintained. Multiple comparison tests are not carried out for results in Tables 2 and 3 due to the lack of statistical difference in convergence time for most of the problems.

In Tables 4 and 5, corresponding results for the GPS and combinatorial problems are shown with binary tournament selection. For the GPS attitude determination problem, through binary local selection, no statistical proof was obtained in terms of convergence time and hit rate for both sets of results. In contrast, the MMDP statistical proof was found in convergence time for $4(\frac{n}{2} \times \frac{n}{2})$ → constant *internal* reconfiguration. However, the best hit rate of 92% ($\sigma = 2.71$) is reached through pre-programmed narrow → square *external* reconfiguration, in comparison to 87% ($\sigma = 3.36$) of former approach. However, the narrow → adaptive genotypic based approach, *external* reconfiguration, provides best efficiency and efficacy.

For the MTTP, high hit rates are in general obtained. There is a statistical difference for the best configuration case among static, pre-programmed and constant approaches: square → constant *internal* reconfiguration. Similarly for adaptive approaches, statistical difference is proved for the best adaptive mechanism: narrow → adaptive genotypic based *external* reconfiguration. After a

Table 7. Convergence time and hit rate results for continuous, FMS and SLE problems through *external* and *internal* adaptive lattice reconfiguration with *anisotropic local selection*.

Adaptive Phenotypic					
***External* Reconfig.**	**Rastrigin**	**Griewank**	**Langerman**	**FMS**	**SLE**
Rectangular→Adapt.	227.12±17.92	420.29±40.40	420.21±47.44	369.57±64.78	569.00±81.42
	100%	86%, 3.46	28%, 4.48	61%, 4.87	31%, 4.62
Narrow→Adapt.	227.13±15.99	425.08±40.74	391.00±41.28	358.77±57.90	566.20±90.81
	100%	84%, 3.66	25%, 4.33	57%, 4.95	25%, 4.33
Internal Reconfig.					
Square→Adapt.	188.91±12.42	384.69±51.54	338.28±77.18	314.35±69.18	474.89±89.32
	100%	94%,2.37	25%,4.33	64%,4.8	37%,4.82
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	189.31±16.21	377.34±46.40	340.12±72.40	314.12±59.51	457.31±78.66
	100%	93%,2.55	33%,4.70	66%,4.73	29%,4.53
Adaptive Genotypic					
***External* Reconfig.**					
Rectangular→Adapt.	202.39±18.42	401.90±44.85	387.22±59.31	328.87±65.23	513.23±90.67
	100%	83%, 3.75	27%, 4.43	63%, 4.82	30%, 4.58
Narrow→Adapt.	202.11±16.62	393.69±49.50	382.73±61.50	332.49±70.22	525.77±85.77
	100%	81%, 3.92	26%, 4.38	59%, 4.91	40%, 4.89
Internal Reconfig.					
Square→Adapt.	192.45±13.42	379.38±49.98	320.90±72.27	306.75±53.67	496.33±85.82
	100%,	94%,2.37	32%,4.66	69%,4.62	45%,4.97
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	186.93±9.56	379.92±49.42	357.44±70.00	304.34±62.57	472.45±103.06
	100%,	94%,2.37	38%,4.85	70%,4.58	37%,4.82
^{3}ANOVA/K-W	+	+	+	+	•

doi:10.1371/journal.pone.0041279.t007

multiple comparison test is performed for both sets of results, no statistical proof was found for the adaptive approach in terms of convergence time. The P-Peaks problem through binary tournament selection presents no statistical difference for either static, pre-programmed, constant or for adaptive approaches in terms of convergence time, and hit rates were in all cases 100%.

The ECC problem presents statistical difference in terms of convergence time in both Tables 4 and 5. The best performance is achieved by genotypic based *external* reconfiguration when evolution starts on a rectangular configuration.

At this stage, an interesting preliminary conclusion indicates that the difference in performance between constant and adaptive approaches is not as substantial as expected. Moreover, in Table 4, the best results, in terms of convergence time, correspond to constant *internal* reconfiguration. However, better efficacy is obtained through static and pre-programmed *external* reconfiguration criteria for SLE, GPS and MMDP problems. Another interesting observation in Table 5, for all problems, is that best convergence times are obtained through adaptive genotypic based *external* reconfiguration.

In general, binary tournament local selection provides a narrow span for selective pressure when internally reconfiguring the grid, in comparison to anisotropic local selection, see Figure 2. However, several advantages in performance have been achieved for some problems with statistical proof.

Anisotropic Local Selection

Results obtained through anisotropic local selection are presented in this subsection. In Tables 6 and 7 results for continuous problems and the FMS problem are presented. Anisotropic local selection with constant $\alpha = 0.8$ presents a more explorative behaviour if compared to binary tournament selection. Moreover, this selection method also makes use of structural properties in cGAs, due to neighbourhood configuration in which the location of individuals for selection is implied.

In both tables, statistical difference is proved for most problems except for the SLE problem in Table 7. Therefore, multiple comparison tests were carried out among most approaches. The Rastrigin function shows statistical difference in convergence time for both sets of results. However, no statistical proof supports this difference between best approaches on each table. In comparison to a static square topology, convergence time consistency is held in constant and adaptive criteria.

The Griewank function is better approached by *internal* reconfiguration, square → constant for best convergence time, and $4(\frac{n}{2}\times\frac{n}{2})$ → constant for best hit rate. Similar results are obtained through adaptive approaches. Genotypic and phenotypic based *internal* reconfiguration also provides the best performance. In terms of the hit rate, 93% and 94% are reached. However, after multiple comparison tests, no statistical proof was obtained between constant and adaptive criteria. Moreover, consistency in

Table 8. Convergence time and hit rate results for GPS and combinatorial problems through static, preprogrammed (*external*) and constant (*internal*) lattice reconfiguration with *anisotropic local selection.*

Static/Preprog.					
***External* Reconfig.**	**GPS**	**MMDP**	**MTTP**	**P-Peaks**	**ECC**
Square	80.73 ± 15.43	568.98 ± 54.01	444.45 ± 36.92	286.19 ± 35.95	548.19 ± 75.75
	76%, 4.27	97%, 1.70	31%, 4.62	99%	100%
Rectangular	148.95 ± 35.42	574.03 ± 47.93	– – – –	274.80 ± 34.54	572.7 ± 80.35
	88%, 3.24	97%, 1.70	– –	100%	100%
Narrow	143.49 ± 35.20	– – – –	– – – –	304.96 ± 37.53	828.41 ± 83.35
	91%, 2.86	– –	– –	100%	100%
Square-Narrow	93.69 ± 31.25	572.75 ± 61.37	462.00 ± 0.00	321.44 ± 38.42	527.36 ± 78.80
	82%, 3.84	77%, 4.20	1%, 0.99	100%	100%
Narrow-Square	84.23 ± 20.01	633.40 ± 31.46	463.90 ± 19.46	349.29 ± 36.80	548.19 ± 75.75
	78%, 4.14	80%, 4.00	11%, 3.12	100%	100%
Constant					
***Internal* Reconfig.**					
Square→Constant	72.76 ± 16.93	550.86 ± 55.63	437.34 ± 65.97	256.77 ± 28.67	216.56 ± 58.83
	78%, 4.14	94%, 2.37	100%	100%	100%
4($\frac{n}{2} \times \frac{n}{2}$)→Constant	90.00 ± 23.70	553.96 ± 51.20	403.00 ± 47.85	261.31 ± 28.61	212.65 ± 65.84
	89%, 3.12	91%, 2.86	87%, 3.36	100%	100%
[3]ANOVA/K-W	+	●	+	+	+

doi:10.1371/journal.pone.0041279.t008

convergence time is kept through *internal* and adaptive *external* mechanisms.

The Langerman function also provides best performance through constant and genotypic *internal* reconfiguration. Similar to the previous function, statistical proof, in convergence time, was not found between the best constant and adaptive approaches. On the other hand, convergence time consistency is slightly lost through *internal* constant and adaptive reconfiguration. In terms of hit rate, square → constant and $4(\frac{n}{2} \times \frac{n}{2})$ → adaptive achieve up to 37% and 38% success rate respectively. This represents a significant improvement in efficacy in comparison to a static square topology, from $\sigma = 4.07$ to $\sigma = 4.82$ and $\sigma = 4.85$ in standard deviations.

The FMS problem is highly epistatic, which means genetic interdependency is strong. Constant *internal* reconfiguration gives the best performance through $4(\frac{n}{2} \times \frac{n}{2})$ → constant mechanism with statistical significant difference. On the other hand, adaptive genotypic *internal* reconfiguration mechanisms provide the best convergence time and hit rate respectively. However, once again no statistical proof was obtained between constant and adaptive reconfiguration techniques. Moreover, best hit rates are 70% and 71% in both cases. Consistency in convergence time is maintained among reconfiguration approaches.

The best convergence time for the SLE problem is achieved through $4(\frac{n}{2} \times \frac{n}{2})$ → constant *internal* reconfiguration together with the highest hit rate of 42%. Statistical difference is proved in the average number of generations with respect to a static square topology, as well as in hit rates where the difference in standard deviation ranges from $\sigma = 3.92$ to $\sigma = 4.93$. In contrast, results for adaptive approaches are not significantly different in convergence time and a best hit rate of 45% is achieved through genotypic based *internal* reconfiguration. Convergence times are consistent among adaptive mechanisms.

It is worth noticing that through static rectangular topologies most of the problems in Table 6 lose performance quality, increasing convergence time and limiting their hit rates to the minimum, except for the Rastrigin function. Moreover, for static narrow lattices, three of the problems do not converge at all. An important observation is that the best performance results are obtained through constant and adaptive *internal* reconfiguration mechanisms.

In Tables 8 and 9, results for the GPS and combinatorial problems are presented. For most of the problems in each table, statistical difference was found in terms of convergence times, except for the MMDP problem in Table 8.

Overall, the GPS problem is the smallest problem. Yet, its landscape is multi-modal and non symmetric also presenting high epistasis. A smaller population size was implemented with 64 individuals and for anisotropic selection, statistical proof was obtained in convergence time. Overall, the best convergence time is achieved through adaptive genotypic based *external* reconfiguration, although a substantially better hit rate of 91% is reached through a static narrow topology. On the other hand, compared with a static square topology, consistency in convergence time is preserved through constant *internal* and adaptive approaches.

The last four problems are combinatorial. Hit rates of 97% are achieved through static square and rectangular topologies for the MMDP problem, while narrow topology cannot converge at all. No statistical proof is obtained in terms of convergence time. For adaptive approaches, the best performance was obtained through phenotypic and genotypic based $4(\frac{n}{2} \times \frac{n}{2})$ → adaptive *internal* reconfiguration with statistical difference in convergence time and 95% success search rate. After the multiple comparison test,

Table 9. Convergence time and hit rate results for GPS and combinatorial problems through *external* and *internal* adaptive lattice reconfiguration with *anisotropic local selection*.

Adaptive Phenotypic					
***External* Reconfig.**	**GPS**	**MMDP**	**MTTP**	**P-Peaks**	**ECC**
Rectangular→Adapt.	75.96±12.25	648.81±35.58	377.73±47.20	290.93±35.31	257.54±66.39
	81%, 3.92	33%, 4.70	95%, 2.17	100%	100%
Narrow→Adapt.	80.03±18.34	659.28±22.11	377.67±38.69	282.52±37.87	253.3±63.59
	79%, 4.07	50%, 5.00	97%, 1.70	100%	100%
Internal Reconfig.					
Square→Adapt.	80.82±17.40	586.48±46.23	405.36±46.17	262.85±37.87	212.71±60.87
	81%, 3.92	91%, 2.86	90%, 3.0	100%,	100%,
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	86.67±21.48	583.50±52.83	408.97±45.63	272.38±36.00	222.51±50.51
	84%, 3.66	95%, 2.17	85%, 3.57	100%,	100%,
Adaptive Genotypic					
***External* Reconfig.**					
Rectangular→Adapt.	84%, 3.66	95%, 2.17	85%, 3.57	100%,	100%,
	71.42±15.30	618.30±47.03	362.03±42.30	239.70±28.24	204.74±53.05
	82%, 3.84	78%, 4.14	99%, 0.99	100%	100%
Narrow→Adapt.	65.75±12.93	623.41±48.18	354.02±47.43	242.20±39.50	209.38±64.29
	78%, 4.14	77%, 4.20	98%, 1.40	100%	100%
Internal Reconfig.					
Square→Adapt.	78%, 4.14	77%, 4.20	98%, 1.40	100%	100%
	85.96±20.22	582.54±48.97	405.49±48.34	265.20±31.75	222.66±59.09
	84%, 3.66	91%, 2.86	83%, 3.75	100%,	100%,
$4(\frac{n}{2}\times\frac{n}{2})$→Adapt.	92.89±24.54	580.21±49.54	413.43±47.29	272.33±33.38	226.98±62.60
	89%, 3.12	92%, 2.71	86%, 3.46	100%,	100%,
[3]ANOVA/K-W	+	+	+	+	+

doi:10.1371/journal.pone.0041279.t009

statistical proof is also found between best convergence times on each table while consistency is also maintained.

The MTTP presents a very low hit rate for a static square population topology. Moreover, through rectangular and narrow topologies, cGAs are not appropriate to solve this problem. However, the best convergence time and hit rate are achieved by constant *internal* reconfiguration and the hit rate is improved up to 100%. Adaptive approaches show, with statistical difference, improvement in terms of convergence time. Adaptive genotypic based *external* reconfiguration outperforms other approaches. Yet, hit rate reaches 98% and 99% in best cases. Convergence time consistency is kept among *internal* constant and adaptive reconfiguration.

Overall, hit rates for the P-Peaks and the ECC problems are 100%. In terms of convergence time, statistical tests prove an improvement. Constant *internal* and adaptive genotypic based *external* reconfiguration outperform the rest of the proposed techniques, although, there is no statistical proof to differentiate between both approaches. The consistency of the results is well maintained between the best approaches in both problems.

Similar to the results in Table 6 for static, pre-programmed and constant approaches; in Table 8 best convergence times and most of the best success search rates correspond to constant *internal* reconfiguration. Having a static topology or to perform a pre-programmed change in population topology during evolution does not provide the best performance cGAs could achieve. However, in some cases better hit rates are obtained through static and pre-programmed reconfiguration approaches.

Considering take-over time analysis carried out in Section, less influence on cGAs' performance was expected when lattice reconfiguration mechanisms with binary tournament selection are applied. In contrast, due to a wider span for controlling selective pressure showed by anisotropic selection when $\alpha = 0.8$, a significant effect in performance was anticipated; as supported by statistical tests carried out for each benchmark problem.

In Table 10, convergence time results for adaptive lattice reconfiguration mechanisms with anisotropic selection are statistically compared. In the second column, the best adaptive approach per problem with statistical significance is included, either best *external* or *internal* lattice reconfiguration criterion. *Internal* adaptive mechanisms outperformed *external* adaptive ones in six out of ten benchmark problems. In columns 3-9, static, pre-programmed and constant reconfiguration mechanisms are statistically compared to best adaptive ones. In most cases, adaptive approaches outperformed static and pre-programmed ones. However, no statistically supported evidence is found for adaptive mechanisms to improve *internal* constant lattice reconfiguration but in three cases (GPS, MTTP and P-Peaks) where genotypic based *external* adaptive approaches outperformed constant ones. Efficacy achieved is very similar when applying

Table 10. Convergence times comparison for best *external* and *internal* adaptive approaches versus pre-programmed, static and constant lattice reconfiguration mechanisms through anisotropic local selection.

	Best adaptive approach	***External* static and pre-programmed**					***Internal* constant**	
		Sqr.	**Rect.**	**Narr.**	**Sqr.-Narr.**	**Narr.-Sqr.**	**Sqr.→Cte.**	**$4(\frac{n}{2}\times\frac{n}{2})\rightarrow$Cte.**
Rastrigin	*Internal* Phenotypic Sqr.→Adapt.	+	+	+	+	+	●	●
Griewank	*Internal* Genotypic Sqr.→Adapt.	+	+	−−	+	+	●	●
Langerman	*Internal* Genotypic Sqr.→Adapt.	+	+	−−	+	+	●	●
FMS	*Internal* Genotypic $4(\frac{n}{2}\times\frac{n}{2})\rightarrow$Adapt.	+	+	−−	+	+	●	●
SLE	*Internal* Genotypic $4(\frac{n}{2}\times\frac{n}{2})\rightarrow$Adapt.	+	+	−−	+	+	●	●
GPS	*External* Genotypic Narr.→Adapt.	+	+	+	+	+	+	+
MMDP	*Internal* Genotypic $4(\frac{n}{2}\times\frac{n}{2})\rightarrow$Adapt.	●	●	●	●	+	●	●
MTTP	*External* Genotypic Narr.→Adapt.	+	−−	−−	+	+	+	+
P-Peaks	*External* Genotypic Rect.→Adapt.	+	●	●	●	●	+	+
ECC	*External* Genotypic Rect.→Adapt.	+	+	+	+	+	●	●

doi:10.1371/journal.pone.0041279.t010

adaptive or constant mechanisms with no statistical significant difference. Next section provides final conclusions of this study.

Discussion

The aim of this research has been to extend previous studies on how structural properties in cGAs can be a factor in adequately tuning diversity during evolution. *Internal* lattice reconfiguration has been experimentally assessed and compared with *external* reconfiguration proposed in [26]. The main difference between both approaches is that individ'uals adjacency is maintained through *internal* lattice reconfiguration while it is lost in the *external* approach because it induces an implicit migration mechanism. Making difficult to distinguish if performance improvements are a consequence of the added effect of the lattice reconfiguration mechanism and the induced migration.

Binary tournament showed a narrow span for modifying selective pressure when *internal* lattice reconfiguration is performed. Thus, a significant effect on selective pressure to tune diversity and consequently in performance was not expected. In contrast, take-over times for anisotropic selection showed a wider span to affect selective pressure when the population's topology is internally reconfigured; for most of the problems statistical difference was proved in terms of convergence time.

Internal adaptive approaches outperformed *external* ones in six out of ten benchmark problems when applying anisotropic local selection. In order to statistically compare the best adaptive criteria, tests were carried out versus static, pre-programmed and constant reconfiguration mechanisms. In three out of ten benchmark problems, adaptive approaches (genotypic based *external* reconfiguration) outperformed constant *internal* reconfiguration.

In cGAs, exploration is carried out globally throughout the grid while exploitation is mostly promoted locally within neighbourhoods. The proposed mechanism for diversity tuning works at a global level through the internal reconfiguration of the population's topology as a way of modifying the induced selective pressure and thus promote a more balanced exploration-exploitation trade-off. Therefore, its effect does not need to be a prompt response to changes in the solutions or the representation spaces; this is why constant reconfiguration of the topology induces a similar effect in performance as measuring phenotypic or genotypic entropy and is less computationally expensive.

Author Contributions

Conceived and designed the experiments: AMR ATE. Performed the experiments: AMR. Wrote the paper: AMR ATE.

References

1. Gordon S, Mathias K, Whitley D (1994) Cellular genetic algorithms as function optimizers: Locality effects. In: proceedings of the 1994 ACM Symposium on Applied Computing. ACM Press, 237–241.
2. Alba E, Troya JM (1999) A survey of parallel distributed genetic algorithms. Complex 4: 31–52.
3. Tomassini M (1999) Parallel and distributed evolutionary algorithms: A review.
4. Turton B, Arslan T, Horrocks D (1994) A hardware architecture for a parallel genetic algorithm for image registration. In: IEE Colloquium on Genetic Algorithms in Image Processing and Vision. 11/1–11/6.
5. Turton B, Arslan T (1995) An architecture for enhancing image processing via parallel genetic algorithms and data compression. In: IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications. 493–498.
6. Evans J, Arslan T (2002) The implementation of an evolvable hardware system for real time image registration on a system-on-chip platform. In: NASA/DoD Conference on Evolvable Hardware. 142–146.
7. Turton B, Arslan T (1995) A parallel genetic vlsi architecture for combinatorial real-time applications - disc scheduling. In: IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications. 493–498.
8. Xu J, Arslan T, Wan D, Wang Q (2002) Gps attitude determination using a genetic algorithm. In: Proceedings of IEEE Congress on Evolutionary Computation, CEC'02. IEEE, 998–1002. URL ieeexplore.ieee.org/.
9. Xu J, Arslan T, Wang Q, Wan D (2002) An ehw architecture for real-time gps attitude determination based on parallel genetic algorithm. In: Proceedings of Conference on Evolvable Hardware NASA/DoD. 133–141.
10. Stefatos E, Arslan T (2004) An efficient fault-tolerant vlsi architecture using parallel evolvable hardware technology. In: Proceedings of NASA/DoD Conference on Evolvable Hardware. IEEE, 97–103.
11. Stefatos E, Arslan T (2004) High-performance adaptive gps attitude determination vlsi architecture. In: IEEE Workshop on Signal Processing Systems, SIPS. IEEE, 233–238.
12. Tomassini M (2005) Spatially Structured Evolutionary Algorithms, Artificial Evolution in Space and Time. Springer. Series: Natural Computing Series.
13. Andraka R (1998) A survey of cordic algorithsm for fpga based computers. ACM, 191–200.

14. Oei C, Goldberg D, Chang SJ (1991) Tournament selection, niching and the preservation of diversity. In: IlliGAL Report No. 91011.
15. Sarma J, Jong KD (1996) An analysis of the effects of neighbourhood size and shape on local selection algorithms. In: Parallel Problem Solving from Nature. Springer, 236–244.
16. Cantu-Paz E (2000) Efficient and Accurate Parallel Genetic Algorithms. Genetic Algorithms and Evolutionary Computation. Kluwer Academic Publishers.
17. Cantu-Paz E (1995) A summary of research on parallel genetic algorithms. In: IlliGAL report 95007, University of Illinois at Urbana-Champaign.
18. Cantu-Paz E (1999) Migration policies and takeover times in parallel genetic algorithms. In: Proceedings of GECCO-99: Genetic and Evolutionary Computation Conference. 775–782.
19. Alba E, Troya J (2002) Improving flexibility and efficiency by adding parallelism to genetic algorithms. In: Statistics and Computing. Kluwer Academic Publishers, 12(2): 91–114.
20. Alba E, Troya JM (1999) An analysis of synchronous and asynchronous parallel distributed genetic algorithms with structured and panmictic islands. In: In IPPS/SPDP Workshops. 248–256.
21. Alba E, Dorronsoro B (2008) Cellular genetic algorithms. In: Operations Research/Computer Science Interfaces. Springer.
22. Alba E, Nebro A, Troya J (2002) Heterogeneous computing and parallel genetic algorithms. In: Journal of Parallel and Distributed Computing. Elsevier Science, 62(9): 1362–1385.
23. Alba E, Luna F, Nebro A (2004) Advances in parallel heterogeneous genetic algorithms for continuous optimization. In: International Journal of Applied Mathematics and Computer Science. AMCS, 14(3): 101–117.
24. Alba E, Tomassini M (2002) Parallelism and evolutionary algorithms. In: IEEE Transactions on Evolutionary Computation. IEEE Press, 6(5): 443–462.
25. Alba E, Troya J (2000) Cellular evolutionary algorithms: Evaluating the influence of ratio. In: Proceeding of the 6th International Conference on Parallel Problem Solving from Nature. LNCS Springer, 29–38.
26. Alba E, Dorronsoro B (2005) The exploration/exploitation tradeoff in dynamic cellular genetic algorithms. In: IEEE Transactions on Evolutionary Computation. IEEE.
27. Giacobini M, Tomassini M, Tettamanzi A, Alba E (2005) Selection intensity in cellular evolutionary algorithms for regular lattices. In: IEEE Transactions on Evolutionary Computation. IEEE, 489–505.
28. Bäck T, Breukelaar R (2005) Using genetic algorithms to evolve behavior in cellular automata. In: Lecture Notes in Computer Sciences 3699. Springer-Verlag, 1–10.
29. Simoncini D, Collard P, Verel S, Clergue M (2007) On the influence of selection operators on performances in cellular genetic algorithms. In: Proceeding of IEEE Congress on Evolutionary Computation (CEC'07). IEEE, p. 47064713.
30. Kohlmorgen U, Schmeck H, Haase K (1996) Experiences with fine-grained parallel genetic algorithms. Annals of Operations Research 90: 203–219.
31. Baluja S (1993) Structure and performance of fine-grain parallelism in genetic search. In: Technical Report, Carnegie Mellon University.
32. Murata T, Takada K (2004) Performance evaluation of a distributed genetic algorithm with cellular structures on function optimization problems. In: Proceedings of 8th International Conference on Knowledge-Based Intelligent Information and Engineering Systems. Springer-Verlag, p. 11281135.
33. Simoncini D, Collard P, Verel S, Clergue M (2006) From cells to islands: An unified model of cellular parallel genetic algorithms. In: 2006 International Conference on Cellular Automata. Springer-Verlag, p. 248257.
34. Muhammad A, Bargiela A, King G (1990) Fine-grained parallel genetic algorithm: A stochastic optimization method.
35. Muhammad A, Bargiela A, King G (1999) Fine-grained parallel genetic algorithm: A global convergence criterion. In: International Journal in Computer Mathematics. Gordon and Breach Science, 139–155.
36. Ortiz-Boyer D, Hervś-Martínez C (2012). School of computer science website, carnegie mellon university, benchmark problems, accessed 2012 june 25th. URL http://www.cs.cmu.edu/afs/cs/project/jair/pub/volume24/ortizboyer05a-html/node6.html.
37. Bersini H, Dorigo M (1996) Results of the first international contest on evolutionary optimisation (1st iceo). In: Proceedings of IEEE International Conference on Evolutionary Computation. IEEE, 611–615.
38. Morales-Reyes A, Stefatos E, Erdogan A, Arslan T (2008) Towards fault-tolerant systems based on adaptive cellular genetic algorithms. In: Proceedings of IEEE NASA/ESA Conference on Adaptive Hardware and Systems (AHS'08). IEEE, 398–405. URL ieeexplore.ieee.org/.
39. Chen H, Member S, Flann NS, Society IC, Watson DW, et al. (1998) Parallel genetic simulated annealing: a massively parallel simd algorithm. IEEE Transactions on Parallel and Distributed Systems 9: 126–136.
40. Morales-Reyes A, Haridas N, Erdogan A, Arslan T (2009) Adaptive fault tolerant gps attitude determination system. In: Proceedings of IEEE Aerospace Conference. IEEE, 1–8. URL ieeexplore.ieee.org/.