Modeling Selection Intensity for Linear Cellular Evolutionary Algorithms

Mario Giacobini[1], Marco Tomassini[1], and Andrea Tettamanzi[2]

[1] Computer Science Institute, University of Lausanne, Switzerland
{mario.giacobini, marco.tomassini}@unil.ch
[2] Information Technologies Department, University of Milano, Italy
andrea.tettamanzi@unimi.it

Abstract. We present quantitative models for the selection pressure on cellular evolutionary algorithms structured as a ring of cells. We obtain results for synchronous and asynchronous cell update policies. Theoretical results are in agreement with experimental values and show that the selection intensity can be controlled by using different update methods.

1 Introduction

Cellular evolutionary algorithms (cEAs) are an example of spatially structured evolving populations that is often used in optimization and other applications [1]. The structure may be an arbitrary graph, but more commonly it is a one-dimensional or two-dimensional grid. This kind of evolutionary algorithm has become popular because it is easy to implement on parallel hardware of the SIMD or MIMD type. Although SIMD machines have almost disappeared except for special purpose computing, cEAs can still be very conveniently implemented on computer clusters with excellent performance gains. However, what really matters is the model, not its implementation. Thus, in this work we will focus on cEA models and on their properties without worrying about implementation issues.

The theory of cEAs is relatively underdeveloped, although several results have been published on selection pressure and convergence speed. Sarma and De Jong performed empirical analyses of the dynamical behavior of cellular genetic algorithms (cGAs) [2, 3]. Their work concentrated on the effect that the local selection method, the neighborhood size, and neighborhood shape have on the selection pressure. Rudolph and Sprave [4] have shown how cGAs can be modeled by a probabilistic automata network and have provided proofs of complete convergence to a global optimum based on Markov chain analysis for a model including a fitness threshold. We have recently studied the selection pressure behavior in cEAs on two-dimensional, torus-shaped grids [5].

Our purpose here is to investigate selection pressure in one-dimensional systems in detail. We study two kinds of dynamical systems: synchronous and asynchronous. For synchronous cEAs, some results are available, such as Sprave's hypergraph model [6], Gorges-Schleuter's study of evolution strategies [7], and

Rudolph's theoretical analysis [8]. We complete these results and extend the investigation to asynchronous linear cEAs, which, to our knowledge, have never been studied before from this perspective. In particular, we would like to be able to model observed takeover-time curves with simple difference equations describing the propagation of the best individual under probabilistic conditions.

Section two introduces linear synchronous and asynchronous cEAs. Next we define the concept of takeover time and the models predicting the selection pressure curves in the synchronous and asynchronous update methods. Empirical results and model accuracy are then discussed comparing the experimental curves to the predicted curves. Section 6 gives our conclusions and the future lines of this research.

2 Linear Cellular Genetic Algorithms

In a linear cEA the individuals (also called cells) are arranged along a line. Depending on whether the last and the first individuals communicate or not we have a ring or an array topology. Here we assume the first case, which is more common. Each individual has the same number of neighbors on both sides, and this number depends on the *radius* r. We will only consider the simplest case, $r = 1$, which means that there are three neighbors, including the central cell itself. Let us call S the (finite) set of the states that a cell can assume: this is the set of points in the (discrete) search space of the problem. The set N_i is the set of neighbors of a given cell i, and let $|N_i| = N$ be its size. The local transition function ϕ can then be defined as:

$$\phi : S^N \to S,$$

which maps the state $s_i \in S$ of a given cell i into another state from S, as a function of the states of the N cells in the neighborhood N_i. In our case, namely a line of cells with $r = 1$, ϕ takes the following form:

$$\phi(\cdot) = P\{x_i(t+1) \mid x_{i-1}(t),\ x_i(t),\ x_{i+1}(t)\},$$

where P is the conditional probability that cell x_i will assume at the next time step $t + 1$ a certain value from the set S, given the current (time t) values of the states of all the cells in the neighborhood. We are thus dealing with probabilistic automata, and the set S should be seen as a set of values of a random variable. The probability P will be a function of the particular selection and variation methods; that is, it will depend on the genetic operators. In this paper we model cEAs using two particular selection methods: binary tournament and linear ranking, but the same framework could easily be applied to other selection strategies.

A cEA starts with the cells in a random state and proceeds by successively updating them using evolutionary operators, until a termination condition is met. Updating a cell in a cellular EA means selecting two parents in the individual's neighborhood, applying genetic operators to them, and finally replacing the individual if an offspring has a better fitness. Cells can be updated *synchronously* or *asynchronously*. In synchronous, or parallel, update all the cells change their states simultaneously, while in asynchronous, or sequential, update cells are updated one at a time in some order.

There are many ways for sequentially updating the cells of a cEA (for a discussion of asynchronous update in cellular automata see [9]). We consider four asynchronous update methods [9]:

- In *fixed line sweep* (LS), the n grid cells are updated sequentially $(1, 2 \dots n)$.
- In *fixed random sweep* (FRS), the next cell to be updated is chosen with uniform probability without replacement; this will produce a certain update sequence $(c_1^j, c_2^k, \dots, c_n^m)$, where c_q^p means that cell number p is updated at time q and $(j, k, \dots, m)$ is a permutation of the n cells. The same permutation is then used for all update cycles.
- The *new random sweep* method (NRS) works like FRS, except that a new random cell permutation is used for each sweep through the array.
- In *uniform choice* (UC), the next cell to be updated is chosen at random with uniform probability and with replacement. This corresponds to a binomial distribution for the update probability.

A *time step* is defined as updating n times sequentially, which corresponds to updating *all* the n cells in the grid for LS, FRS and NRS, and possibly less than n different cells in the uniform choice method, since some cells might be updated more than once.

3 Takeover Time

To study the induced selection pressure without introducing the perturbing effect of variation operators, a standard technique is to let selection be the only active operator, and then measure the time it takes for a single best individual to conquer the whole population i.e., the takeover time [10]. A shorter takeover time thus means a higher selection pressure. Takeover times have been derived by Deb and Goldberg [10] for panmictic populations and for the standard selection methods. These times turn out to be logarithmic in the population size, except in the case of proportional selection, which is a factor of n slower, where n is the population size.

It has been empirically shown in [2] that when we move from a panmictic to a square grid population of the same size with synchronous updating of the cells, the selection pressure induced on the entire population is weaker.

A study on the selection pressure in the case of ring and array topologies in one dimensional cEAs has been done by Rudolph [8]. Abstracting from specific selection methods, he splits the selection procedure into two stages: in the first stage an individual is chosen in the neighborhood of each individual, and then, in the second stage, for each individual it is decided whether the previously chosen individual will replace it in the next time step. Rudolph derives the expected takeover times for the two topologies as a function of the population size and the probability that in the selection step the individual with the best fitness is selected in the neighborhood.

Following Rudolph's hypothesis of non-extinctive selection methods, in this paper we study in detail the one-dimensional case. Moreover, we obtain results not only in the synchronous update case, but also for the asynchronous models presented in the previous section. Models for the growth of the best individual in the form of difference equations are presented in the next section. These models will then be compared with the experimental results in Section 5.

4 Models

Let us consider the random variables $V_i(k) \in \{0,1\}$ indicating the presence in cell $i\,(1 \leq i \leq n)$ of a copy of the best individual ($V_i(k) = 1$) or of a worse one ($V_i(k) = 0$) at time step k, where n is the the population size. The random variable

$$N(k) = \sum_{i=1}^{n} V_i(k)$$

denotes the number of copies of the best individual in the population at time step k. Initially $V_i(1) = 1$ for some individual i, and $V_j(1) = 0$ for all $j \neq i$.

Following Rudolph's definition [8], if the selection mechanism is non-extinctive, the expectation $E[T]$ with $T = \min\{k \geq 1 : N(k) = n\}$ is called the takeover time of the selection method. In the case of spatially structured populations the quantity $E_i[T]$, denoting the takeover time if cell i contains the best individual at time step 1, is termed the takeover time with initial cell i. Assuming a uniformly distributed emergence of the best individual among all cells, the takeover time is therefore given by

$$E[T] = \frac{1}{n}\sum_{i=1}^{n} E_i[T].$$

In the following sections we give the recurrences describing the growth of the random variable $N(k)$ in a cEA with ring topology for the synchronous and the four asynchronous update policies described in Section 2. We consider a non-extinctive selection mechanism that selects the best individual in a given neighborhood with probability $p \in (0,1)$.

4.1 Synchronous Takeover Time

In a synchronous cEA, at each time step k the expected number of copies $N(k)$ of the best individual is independent from its initial position. Since we consider neighborhoods of radius 1, the set of cells containing a copy of the best individual will always be a connected region of the ring. Therefore at each time step, only two more individuals (the two adjacent to the connected region of the ring) will contain a copy of the best individual with probability p. The growth of the quantity $N(k)$ can be described by the following recurrence:

$$\begin{cases} N(0) = 1, \\ E[N(k)] = \sum_{j=1}^{n} P[N(k-1) = j]\ (j + 2p). \end{cases}$$

Since $\sum_{j=1}^{n} P[N(k-1) = j] = 1$, and the expected number $E[N(k-1)]$ of copies of the best individual at time step $k-1$ is by definition $\sum_{j=1}^{n} P[N(k-1) = j]\ j$, the previous recurrence is equivalent to

$$\begin{cases} N(0) = 1, \\ E[N(k)] = E[N(k-1)] + 2p. \end{cases}$$

The closed form of this recurrence is $E[N(k)] = 2pk + 1$, therefore the expected takeover time $E[T]$ for a synchronous ring cEA with n individuals is

$$E[T] = \frac{1}{2p}\ (n-1).$$

Rudolph [8] gave analytical results for the ring with synchronous update only and for a generic probability of selection p. Although obtained in a different way, the previous expression and his equation (2) give nearly the same results for large population sizes n. In fact, his equation, for large n, reduces to $\frac{n}{2p} - \frac{1}{4}$, while our equation gives $\frac{n}{2p} - \frac{1}{2p}$. Given that the first term quickly dominates the second for large n, the two expressions are equivalent.

4.2 Asynchronous Fixed Line Sweep Takeover Time

Let us consider the general case of an asynchronous fixed line sweep cEA, in which the connected region containing the copies of the best individual at time step k is $B(k) = \{r, \ldots, s\}$, $1 < r \leq s < n$. At each time step the cell $r-1$ will contain a copy of the best individual with probability p, while the cells $s+j$ (with $j = 1, \ldots, n-s$) will contain a copy of the best individual with probability p^j. The recurrence describing the growth of the random variable $N(k)$, is therefore

$$\begin{cases} N(0) = 1, \\ E[N(k)] = \sum_{j=1}^{n} P[N(k-1) = j]\ \left(j + p + \sum_{i=1}^{n-j} p^i\right). \end{cases}$$

Since $\sum_{i=1}^{n-j} p^i$ is a geometric progression, for large n we can approximate this quantity by the limit value $p/(1-p)$ of the summation. The recurrence is therefore equivalent to the following one:

$$\begin{cases} N(0) = 1, \\ E[N(k)] = E[N(k-1)] + p + \frac{p}{1-p} = E[N(k-1)] + \frac{1}{1-p}. \end{cases}$$

The closed form of the previous recurrence being

$$E[N(k)] = \frac{1}{1-p}\,k + 1,$$

we conclude that the takeover time for an asynchronous fixed line sweep cEA with a population of size n is

$$E[T] = (1-p)\,(n-1).$$

4.3 Asynchronous Fixed and New Random Sweep Takeover Time

The mean behaviors of the two asynchronous fixed and new random sweep update policies among all the possible permutations for the sweeps are equivalent. We therefore give only one model describing the growth of the random variable $N(k)$ for both policies.

Let us again consider the general case in which the connected region containing the copies of the best individual at time step k is $B(k) = \{r, \ldots, s\}$ (with $1 < r \leq s < n$). The cells $r-1$ and $s+1$ have a probability p of containing a copy of the best individual at the next time step. Because of symmetry reasons, we consider only the part of the ring at the right side of the connected region. The cell $s+2$ has a probability $1/2$ to be contained in the set of cells after cell $s+1$ in the sweep, so it has a probability $(p/2)p$ to contain a copy of the best individual in the next time step. In general, a cell $s+j+1$ has a probability $1/2$ to be after cell $s+j$ in the sweep, so it has a probability $(p/2)^j p$ to contain a copy of the best individual in the next time step. The recurrence describing the growth of the random variable $N(k)$, is therefore

$$\begin{cases} N(0) = 1, \\ E[N(k)] = \displaystyle\sum_{j=1}^{n} P[N(k-1) = j]\,\left(j + 2\sum_{i=1}^{n-j} p\left(\frac{p}{2}\right)^{i-1}\right), \end{cases}$$

which can be transformed into the recurrence

$$\begin{cases} N(0) = 1, \\ E[N(k)] = \displaystyle\sum_{j=1}^{n} P[N(k-1) = j]\,\left(j + 4\sum_{i=1}^{n-j} \left(\frac{p}{2}\right)^{i}\right). \end{cases}$$

Since $\sum_{i=1}^{n-j}(p/2)^i$ is a geometric progression, for large n we can approximate this quantity by the limit value $p/(2-p)$ of the summation. The recurrence is thus equivalent to the following one:

$$\begin{cases} N(0) = 1, \\ E[N(k)] = E[N(k-1)] + \frac{4p}{2-p}. \end{cases}$$

The closed form of the previous recurrence being

$$E[N(k)] = \frac{4p}{2-p}\,k + 1,$$

we conclude that the expected takeover time for an fixed (or new) random sweep asynchronous cEA with a population of size n is

$$E[T] = \frac{2-p}{4p}\,(n-1).$$

4.4 Asynchronous Uniform Choice Takeover Time

To model takeover time for asynchronous uniform choice cEAs it is preferable to use cell update steps u instead of time steps in the recurrences. As for the other update policies the region containing the copies of the best individual at update step u is a connected part of the ring $B(u) = \{r, \ldots, s\}$ (with $1 < r \leq s < n$). At each update step the two cells $r-1$ and $s-1$ have probability $1/n$ to be selected, and each cell has a probability p, if selected, to contain a copy of the best individual after the selection and the replacement phases. The recurrence describing the growth of the random variable $N(u)$, counting the number of copies of the best individual at update step u thus becomes:

$$\begin{cases} N(0) = 1, \\ E[N(u)] = \sum_{j=1}^{n} P[N(u-1) = j] \left(j + 2\,\frac{1}{n}\,p \right), \end{cases}$$

which can be transformed into

$$\begin{cases} N(0) = 1, \\ E[N(u)] = E[N(u-1)] + 2\,\frac{1}{n}\,p. \end{cases}$$

We can easily derive the closed form of the previous recurrence:

$$E[N(u)] = \frac{2}{n}\,p\,u + 1.$$

Since a time step is defined as n update steps, where n is the population size, the expected takeover time for an uniform choice asynchronous cEA is

$$E[T] = \frac{1}{2p}\,(n-1).$$

We notice that the expected takeover time for a uniform choice asynchronous cEA is equal to the expected takeover time for a synchronous cEA.

It should be noted that the present asynchronous uniform choice update model is very similar to what goes under the name of *nonlinear voter model* in the probability literature [11].

5 Empirical Results

Since cEAs are good candidates for using selection methods that are easily extensible to small local pools, we use binary tournament and linear ranking in our experiments. Fitness-proportionate selection could also be used but it suffers from stochastic errors in small populations, and it is more difficult to model theoretically since it requires knowledge of the fitness distribution function. The cEA structure has ring topology of size 1024 with neighborhood of radius 1. Only the selection operator is active: for each cell it selects one individual in the cell neighborhood (the cell and its two adjacent cell at its right and at its left), and the selected individual replaces the old individual only if it has a better fitness.

5.1 Binary Tournament Selection

We have used the binary tournament selection mechanism described by Rudolph [8]: two individuals are randomly chosen with replacement in the neighborhood of a given cell, and the one with the better fitness is selected for the replacement phase.

Figure 1 shows the growth curves of the best individual for the synchronous and the four asynchronous update methods. We can see how, as the models derived in the previous section predict, the mean curves for the synchronous and the asynchronous uniform choice cases are superposed. Also the mean curves for the two asynchronous fixed and new random sweep show a very similar behavior. The graph shows that the asynchronous update methods give an emergent selection pressure greater than that of the synchronous case, growing from the uniform choice to the line sweep, with the fixed and new random sweep in between.

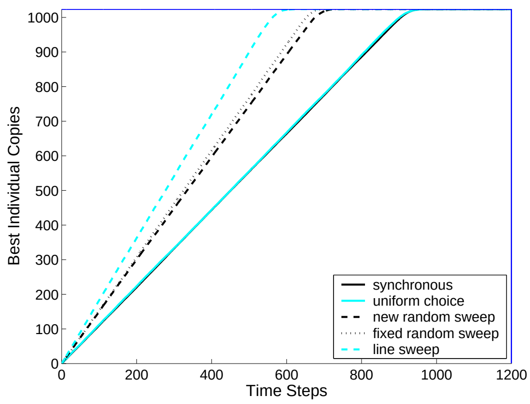

Fig. 1. Takeover times with binary tournament selection: mean values over 100 runs. The vertical axis represents the number of copies $N(k)$ of the best individual in each population as a function of the time step k.

The numerical values of the mean takeover times for the five update methods, together with their standard deviations are shown in Table 1, where it can be seen that the fixed random sweep and new random sweep methods give results that are statistically indistinguishable. The same can be said for the synchronous and the uniform choice methods.

	Synchro	LS	FRS	NRS	UC
Mean Takeover Time	925.03	569.82	666.18	689.29	920.04
Standard Deviation	20.36	24.85	17.38	20.27	26.68

Table 1. Mean takeover time and standard deviation of the tournament selection for the five update methods.

Since we use a neighborhood of radius 1, at most one individual with the best fitness will be present in the neighborhood of a considered cell, except for the last update when there are two of them. It turns out that the probability for an individual having a copy of the best individual in its neighborhood to select it is equal to $p = 5/9$. Using this probability in the models described in Section 3, we calculated the theoretical growth curves. Figure 2 shows the predicted and the experimental curves for the five update methods, and the mean square error between them.

Looking at the curves, it is clear that the models faithfully predict the observed takeover times. Moreover, the equivalence between new random sweep and fixed random sweep, as well as that of synchronous and uniform choice are fully confirmed. The only model presenting a slight discrepancy between theory and experiment is the line sweep. We intend to investigate the source of this error further.

5.2 Linear Ranking Selection

We have used a standard linear ranking selection mechanism. The three individuals in the neighborhood of a considered cell are ranked according to their fitnesses: each individual then has probability $(s - i)/s$ to be selected for the replacement phase, where s is the number of cells in the neighborhood ($s = 3$ in our case) and i is its rank in the neighborhood.

Figure 3 shows the growth curves of the best individual for the synchronous and the four asynchronous update methods. We can observe in the linear ranking case the same behavior that emerged in the binary tournament case: the mean curves for the synchronous and the asynchronous uniform choice cases are superposed, and the mean curves for the two asynchronous fixed and new random sweep show very similar behaviors. The graph shows that the asynchronous update methods give an emergent selection pressure greater than that of synchronous one, growing from the uniform choice to the line sweep, with the fixed and new random sweep in between. The numerical values of the mean takeover times for the five update methods, together with their standard deviations are shown in Table 2. Again, the results show that the two random sweep methods

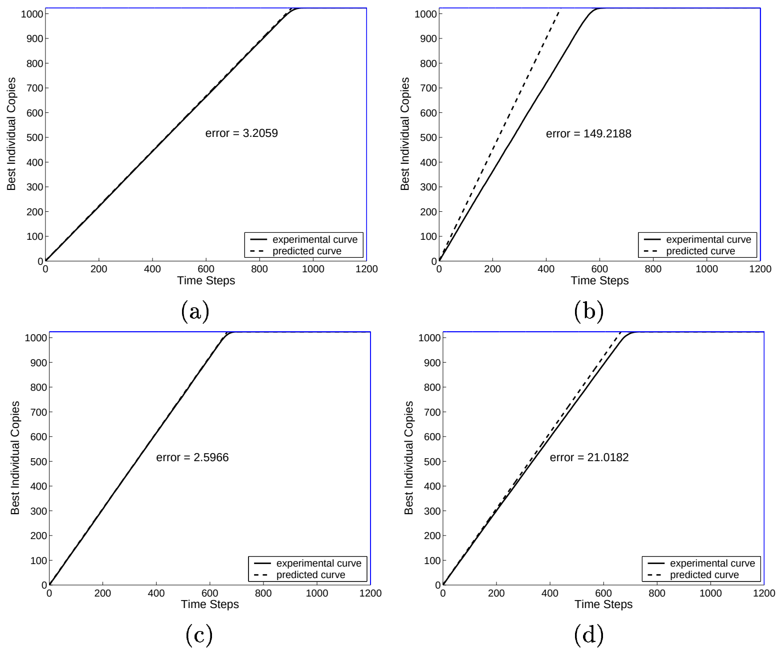

Fig. 2. Comparison of the experimental takeover time curves (full) with the model (dashed) in the case of binary tournament selection for four update methods: synchronous (a), asynchronous line sweep (b), asynchronous fixed random sweep (c), asynchronous new random sweep (d). Asynchronous uniform choice gives the same curve as the synchronous update, therefore it is omitted.

are statistically equivalent, which is also the case for the synchronous and uniform choice methods.

With this linear ranking selection method, a cell having a copy of the best individual in its neighborhood has a probability $p = 2/3$ of selecting it. Using this value in the models described in Section 3, we can calculate the theoretical growth curves. Figure 4 shows the predicted and the experimental curves for the five update methods, and the mean square error between them. The agreement between theory and experiment is very good.

6 Conclusions and Future Work

We have presented quantitative models for the takeover time in cellular evolutionary algorithms structured as a ring with nearest neighbor interactions only.

	Synchro	LS	FRS	NRS	UC
Mean Takeover Time	768.04	387.09	519.92	541.14	766.5
Standard Deviation	17.62	19.21	14.26	14.48	25.44

Table 2. Mean takeover time and standard deviation of the linear ranking selection for the five update methods.

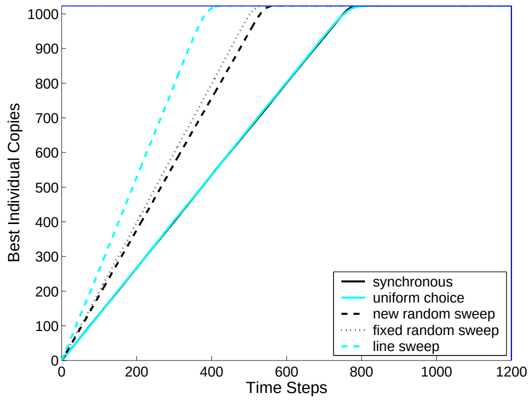

Fig. 3. Takeover times with linear ranking selection: mean values over 100 runs. The vertical axis represents the number of copies $N(k)$ of the best individual in each population as a function of the time step k.

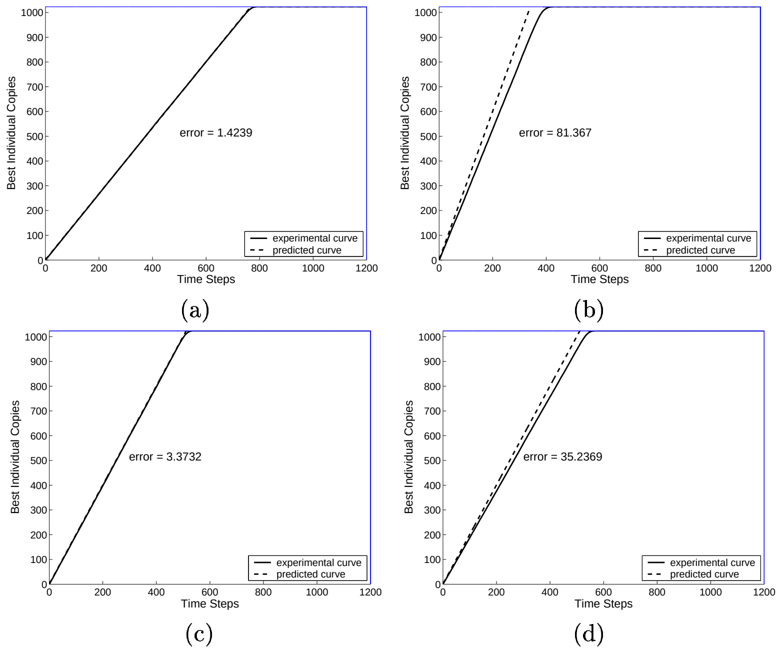

Fig. 4. Comparison of the experimental takeover time curves (full) with the model (dashed) in the case of linear ranking selection for four update methods: synchronous (a), asynchronous line sweep (b), asynchronous fixed random sweep (c), asynchronous new random sweep (d). Asynchronous uniform choice gives the same curve as the synchronous update, therefore it is omitted.

New results have been obtained for asynchronous cell update policies. The models are based on simple difference probabilistic equations. We have studied two types of selection mechanisms that are commonly used in cEAs: binary tournament and linear ranking. With these selection methods, our results show that there is a good agreement between theory and experiment; in particular, we showed that asynchronous cell update methods permit to control the selection intensity in an easy and principled way, without using ad hoc parameters.

In the future, we intend to extend this type of analysis to larger neighborhoods, and to more complex topologies such as two and three-dimensional grids, and to general graph structures. Moreover, we intend to investigate Markov chain modeling of our system and the relationships that may exist with probabilistic particle systems such as voter models.

References

1. Gorges-Schleuter, M.: On the power of evolutionary optimization at the example of ATSP and large TSP problems. In Husbands, P., Harvey, I., eds.: Fourt European Conference on Artificial Life, Cambridge, Massachusetts, The MIT Press (1997)
2. Sarma, J., Jong, K.A.D.: An analysis of the effect of the neighborhood size and shape on local selection algorithms. In Voigt, H.M., Ebeling, W., Rechenberg, I., Schwefel, H.P., eds.: Parallel Problem Solving from Nature (PPSN IV). Volume 1141 of Lecture Notes in Computer Science., Springer-Verlag, Heidelberg (1996) 236–244
3. Sarma, J., Jong, K.A.D.: An analysis of local selection algorithms in a spatially structured evolutionary algorithm. In Bäck, T., ed.: Proceedings of the Seventh International Conference on Genetic Algorithms, Morgan Kaufmann (1997) 181–186
4. Rudolph, G., Sprave, J.: A cellular genetic algorithm with self-adjusting acceptance thereshold. In: In First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, IEE, London (1995) 365–372
5. Giacobini, M., Alba, E., Tomassini, M.: Selection intensity in asynchronous cellular evolutionary algorithms. In et al., E.C.P., ed.: Proceedings of the genetic and evolutionary computation conference GECCO'03, Springer Verlag, Berlin (2003) 955–966
6. Sprave, J.: A unified model of non-panmictic population structures in evolutionary algorithms. In: Congress on Evolutionary Computation (CEC'99), IEEE Press, Piscataway, NJ (1999) 1384–1391
7. Gorges-Schleuter, M.: An analysis of local selection in evolution strategies. In: Genetic and evolutionary conference, GECCO99. Volume 1., Morgan Kaufmann, San Francisco, CA (1999) 847–854
8. Rudolph, G.: On takeover times in spatially structured populations: Array and ring. In et al., K.K.L., ed.: Proceedings of the Second Asia-Pacific Conference on Genetic Algorithms and Applications, Global-Link Publishing Company (2000) 144–151
9. Schönfisch, B., de Roos, A.: Synchronous and asynchronous updating in cellular automata. BioSystems **51** (1999) 123–143
10. Goldberg, D.E., Deb, K.: A comparative analysis of selection schemes used in genetic algorithms. In Rawlins, G.J.E., ed.: Foundations of Genetic Algorithms, Morgan Kaufmann (1991) 69–93
11. Durrett, R.: Lecture Notes on Particle Systems and Percolation. Wadsworth, Belmont CA (1988)