

Spatially Structured Evolutionary Algorithms

Marco Tomassini

University of Lausanne, Switzerland

marco.tomassini@hec.unil.ch

June 27, 2004

Why Topology Matters

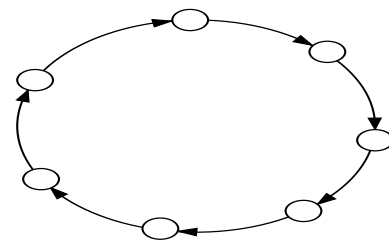
The spatial structure of a population will be called its *topology*

- Population topology has a marked influence on the dynamical processes taking place in the population
- To some extent, the dynamics can be controlled by using the appropriate topology
- Population topology can be mathematically characterized using the tools of graph theory

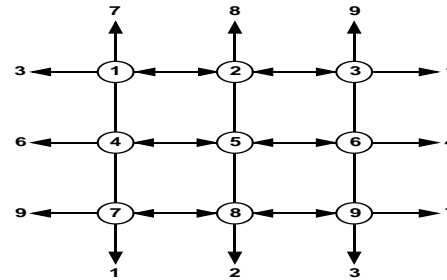
Main Categories of Population Topologies

- Multiple Populations, also called *island* models (each node of the graph is a population in itself)
- Cellular Populations (each node of the graph is a single individual)
- There are many possible hybrid models, such as islands of cellular populations, or islands that themselves contain other islands etc.

Examples of Island Population Topologies

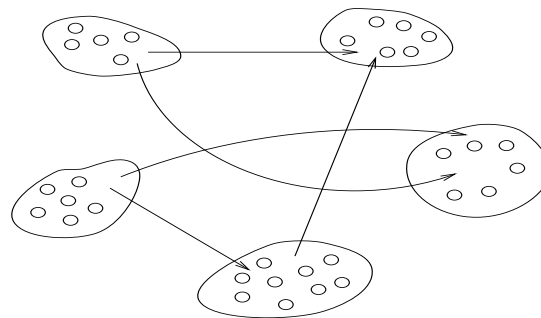


a)



b)

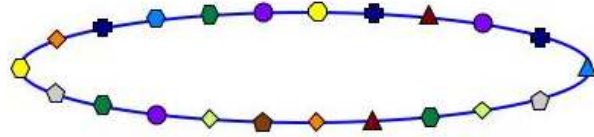
Mesh and Ring Topologies. Each circle represents a panmictic population.



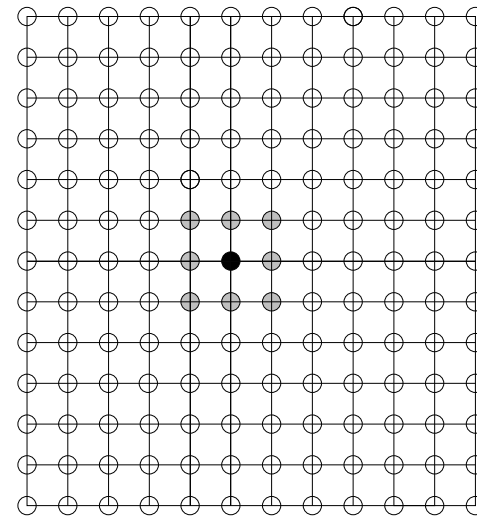
Random Topology

Cellular or Lattice Topologies

Each individual occupies a cell in a 1-D, 2-D or 3-D lattice, or another graph structure



ring cellular structure



grid cellular structure

Evolutionary Algorithms in Structured Populations

Island Models

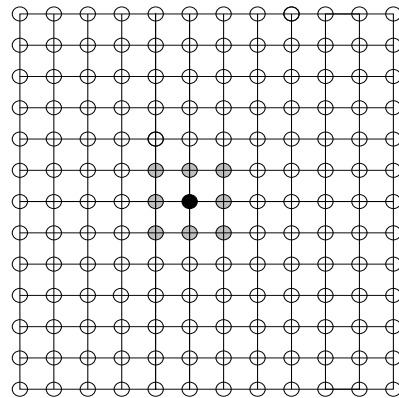
- The whole population is subdivided into a number of subpopulations
- Subpopulations are loosely coupled: they evolve independently for a while
- A topological pattern of communication is established among the islands
- From time to time selected individuals are exchanged between populations and replace local individuals

A number of parameter values must be determined somehow: number of islands (subpopulation size), topology of communication, frequency of migration, individual replacement policy... Some of those might even change during the run

Evolutionary Algorithms in Structured Populations

Cellular Models

- Each individual occupies a cell in a regular lattice or a more general graph
- Genetic operators are local. Selection, mutation and recombination take place only within a small neighborhood.
- After selection and variation, each cell is replaced, e.g., by the best individual in the neighborhood



Case Study: Selection Pressure in Cellular EAs

It is a good case study because:

- The effects of topology are most easily seen in cellular EAs
- Selection pressure is a fundamental aspect of EAs
- Variation operators do not interfere with the dynamics
- The mathematical analysis is possible in some cases

Selection Pressure and Takeover Times

Takeover Time is the time it takes for a single best individual to take over the whole population

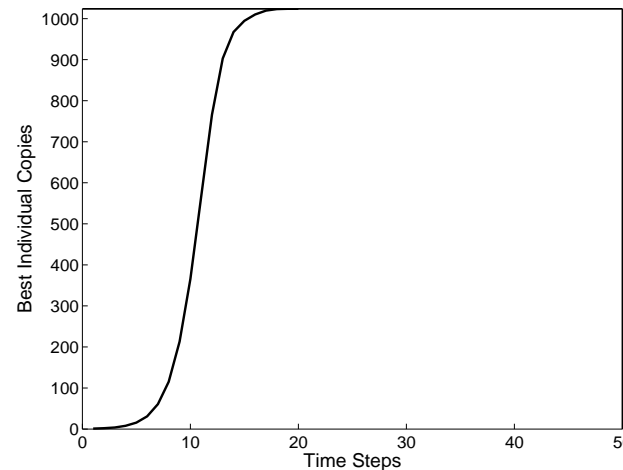
No variation operators: only **selection** is active with a probability p_s that depends on the selection method

Long takeover times mean less intense selection and viceversa for short TT

Selection intensity is related to the **explorative** or **exploitative** character of an EA: the stronger the selection the more exploitative the EA

Growth Curves in Panmictic Populations

In mixing populations the best individual propagates under selection following a *Logistic Curve*.



Analytical and experimental results indicate that, among the usual selection methods, (μ, λ) , tournament and ranking induce a stronger selection pressure than fitness proportionate selection

The Origins of Logistic Growth

Logistic growth occurs in situations where the growth is exponential at first but then it flattens out being limited by diminishing “resources”. In our case, it means that, as time goes by, less and less individuals remain to be “conquered”

Thus, the growth rate is not simply proportional to the current amount N , but rather to a maximum possible “capacity” θ , minus the current amount (Verhulst):

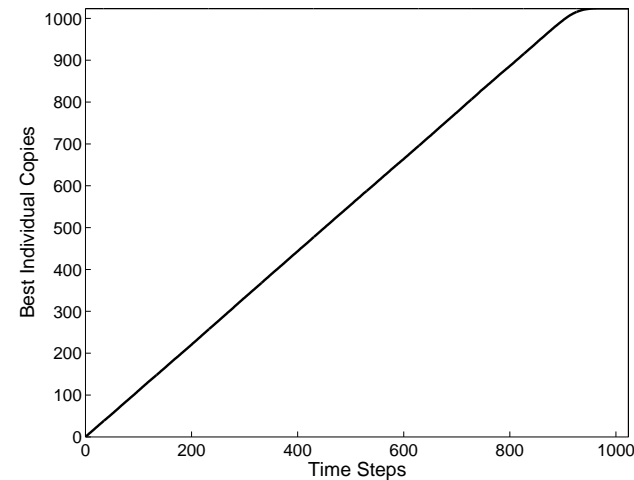
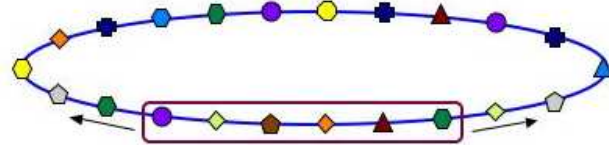
$$\frac{dN}{dt} = \alpha N(\theta - N)$$

which has the solution:

$$N(t) = \frac{N(0)e^{\alpha t}}{1 - N(0)(1 - e^{\alpha t})}$$

Growth Curves in Rings

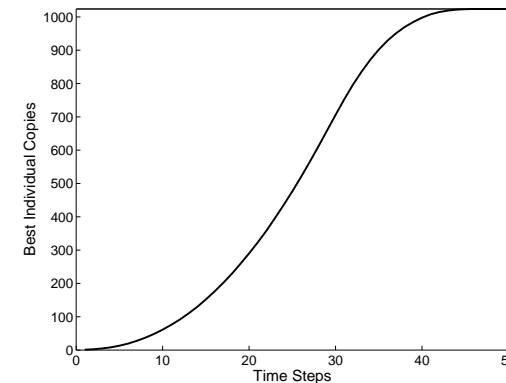
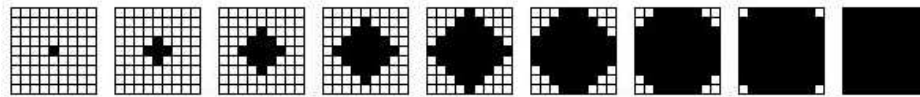
In rings the best individual can only grow at a *linear* rate:



The frontier of the growing region can only expand, at best, to the next two individuals on the next time step

Growth Curves in Two-Dimensional Lattices

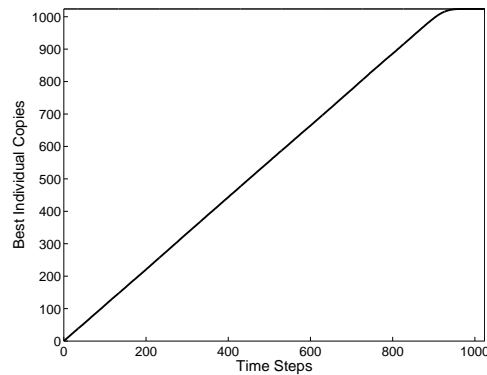
In grids the best individual can only grow at most at a *quadratic* rate:



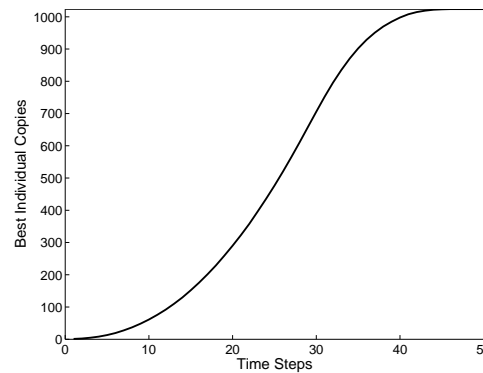
The *diameter* of the expanding region grows at a linear rate, and thus the whole area, which is proportional to the population size, grows at *quadratic* rate

Growth Curves and Topology

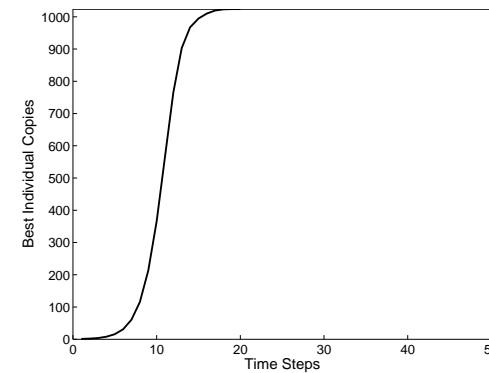
The influence of the population structure is clearly seen:



ring



torus



panmictic

The growth rate, and thus the selection pressure, is much slower in rings than it is in 2-D grids, which is in turn slower than the mixing population

Mathematical Models for Growth Curves I

Our models are based on probabilistic difference equations. The general recurrence for the *expectation* of $N(t)$ for *synchronous* dynamics is:

$$E[N(t)] = \sum_{i=1}^n P[N(t-1) = i] \left(i + \sum_{r=1}^{n-i} \sum_{j=1}^{n-1} P[K = j] \sum_{l=0}^j P[B_j = l] p_s(j, l) \right)$$

where $N(t)$ is a random variable denoting the number of copies of the best individual at time t ; $N(t-1) = i$ is this number at time $t-1$, K is the number of neighbors of a given individual, B_j is the number of copies of the best among the j neighbors of an individual, and $p_s(j, l)$ is the probability of selecting a best among the h best of the j neighbors. The P s denote probabilities.

Mathematical Models for Growth Curves II

- The previous equation is valid for any topology. However, it can be exactly solved only in the linear lattice (ring) case [8]. For other topologies, approximations must be made, and the recurrences cannot, in general, be given in closed form

- For *rings* with a neighborhood of three individuals, the solution is:

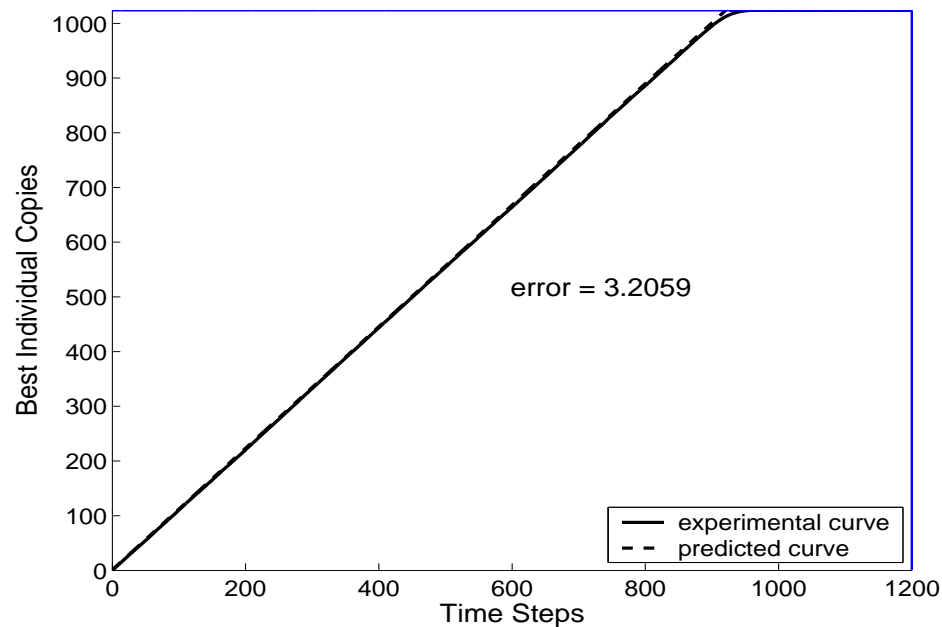
$$E[N(t)] = 2p_s t + 1$$

where the actual probability p_s should be inserted for different selection methods

- The equation can easily be checked for the deterministic case $p_s = 1$ in which $N(t)$ is no longer an expectation (i.e. a random variable)

Comparing Theory and Experiments II

As expected, for the ring case the agreement between theory and experiment is excellent. The experimental curve (black) is the average of 100 runs. Selection method: binary tournament. Population size is 1024.



Mathematical Models for Growth Curves III

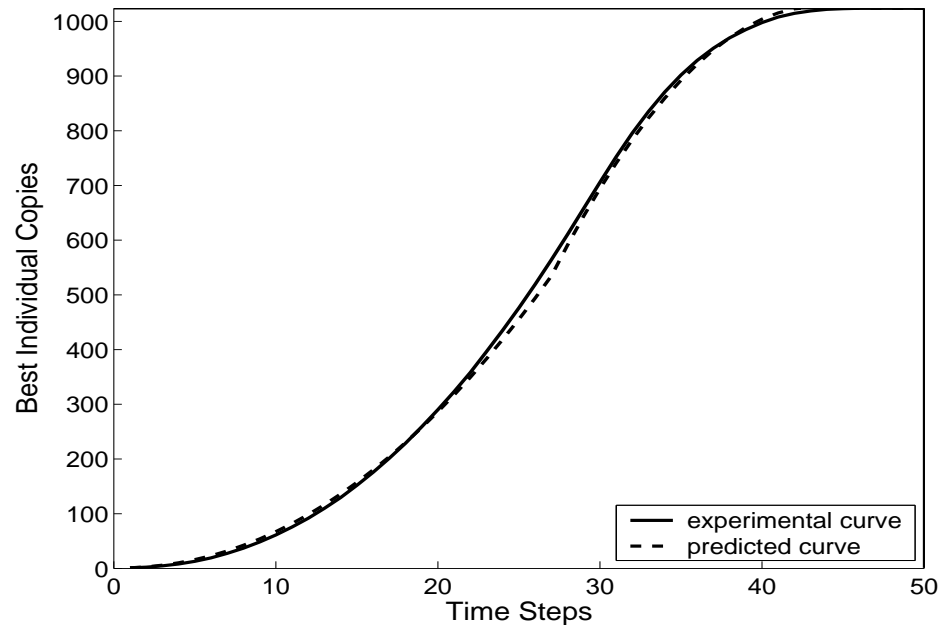
- For the synchronous growth curve in a 2-D torus, assuming a 5 cell (NWCES) neighborhood we get:

$$\left\{ \begin{array}{l} N(0) = 1 \\ N(t) = N(t-1) + 4p_2 \frac{\sqrt{N(t-1)}}{\sqrt{2}} \quad , \quad \text{for } N(t) \leq \frac{n}{2} \\ N(t) = N(t-1) + 4p_2 \sqrt{n - N(t-1)} \quad , \quad \text{for } N(t) > \frac{n}{2} \end{array} \right.$$

- The approximation is geometrical and is based on the growth of a closed planar shape that contains the region of interest (a 45 degrees rotated square). p_2 is the selection-dependent probability of selecting the best individual when there are two copies of it in the neighborhood [9]

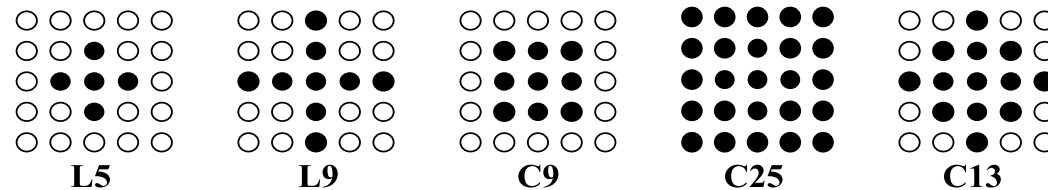
Comparing Theory and Experiments III

For the torus case the agreement between theory and experiment is still good, in spite of the approximations in the model. The experimental curve (full) is the average of 100 runs. Selection is by binary tournament. Population size is 1024.



What About the Neighborhood?

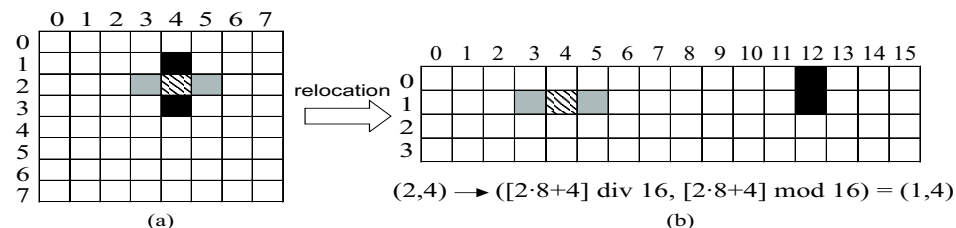
- What happens if the neighborhood's *size* and *shape* change?
- It would be easy to modify the model to take that into account. However, the effects had already been empirically studied by Sarma and De Jong for 2-D grids [11,12]
- Their conclusion: propagation times, and thus selection pressure, are closely related to the neighborhood's size. *Larger* neighborhoods imply *stronger* selection pressure



- Also: neighborhoods having the same “linear extension” such as L9 and C13 induce a similar selection pressure; thus, neighborhood's *shape* matters too

Neighborhood Size and Shape: the Ratio

- Sarma and De Jong were able to characterize the global induced selection pressure by a single parameter: the *ratio* r
- The ratio is, in essence, the radius of a circle centered on the mean center (\bar{x}, \bar{y}) of a neighborhood pattern of n points
- Under this measure $r(L9) = 1.49$ and $r(C13) = 1.47$, which explains why the selection pressure is similar
- As the ratio \rightarrow size of the grid, selection pressure \rightarrow panmictic
- Alba and Troya later extended the concept of ratio to take into account the *whole grid shape*
- Selection pressure *decreases* as the grid *flattens*



Mathematical Models for Growth Curves IV

The last two cases are the usual *panmictic* population, and the *random graph* structure

- A random graph with n vertices can be constructed by taking all possible pairs of vertices and connecting each pair with probability q , or not connecting it with probability $1 - q$
- A panmictic population can be seen as a completely connected graph or, equivalently, as a random graph with probability $q = 1$ of having an edge between any pair of vertices; such a graph has thus $\frac{1}{2}n(n - 1)$ edges.

Mathematical Models for Growth Curves V

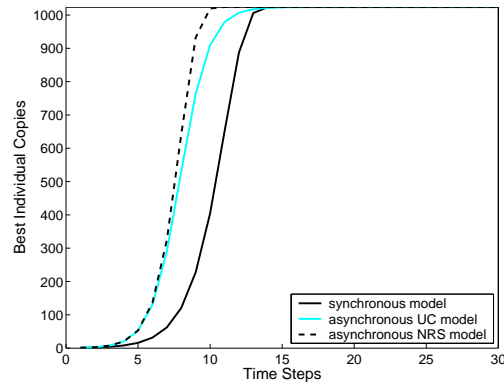
- In the completely connected graph (i.e. panmictic population), the number of neighbors of any individual is $n - 1$
- The random graph case is difficult to solve, since the number of neighbors (i.e. vertex degree) of a given vertex is a binomially distributed random variable. However, the *mean degree* is a constant equal to $q(n - 1)$. We thus use the *mean-field hypothesis*, taking for all individuals the same average number of neighbors
- We only consider *connected* RGs. Disconnected components do not make sense here

Mathematical Models for Growth Curves VI

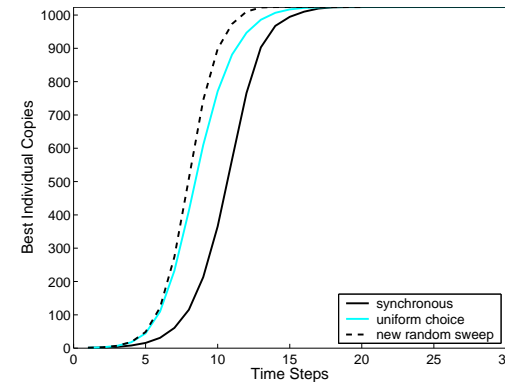
With the mean-field approximation, it turns out that both the *panmictic* and *random graph* topologies obey the *same* growth equation. The growth is obviously logistic in form, and is given as a discrete recurrence:

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

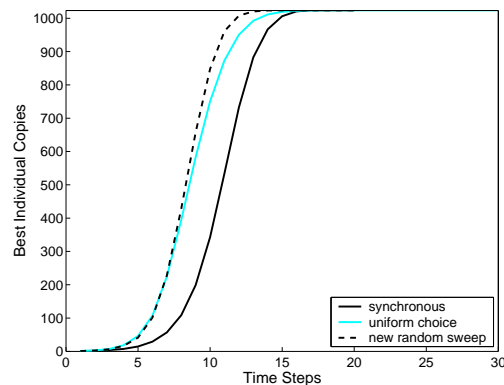
Comparing Theory and Experiments IV



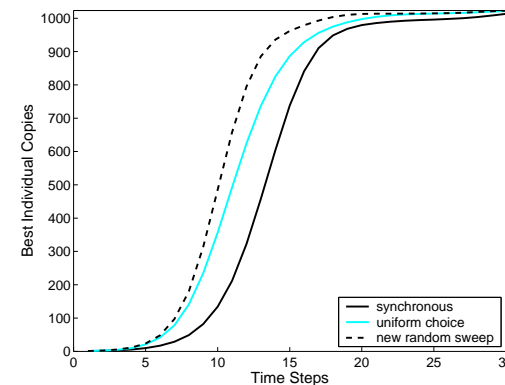
theoretical



panmictic



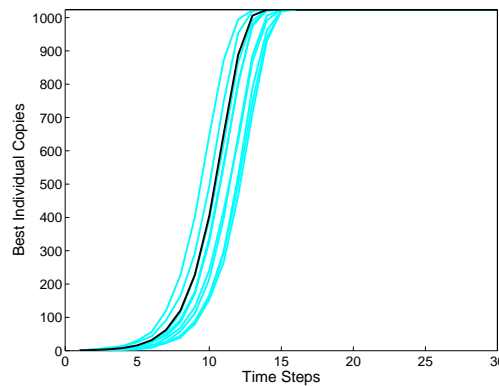
random graph $q=0.1$



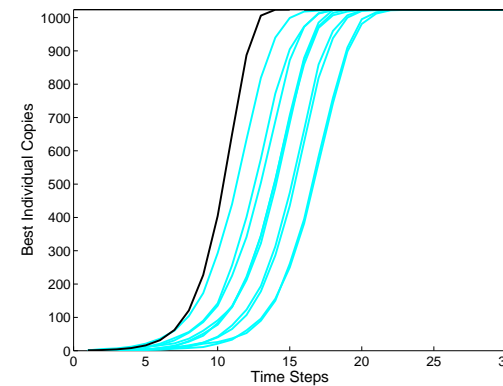
random graph $q=0.01$

Comparing Theory and Experiments V

The agreement between theory (full curve) and experiment (light curves) is very good for the random graph with $q = 0.1$:



$q = 0.1$



$q = 0.01$

The fit is bad for small q . This is due to the mean-field approximation: for $n = 1024$ the average number of neighbors is ~ 100 for $q = 0.1$, while it is ~ 10 for $q = 0.01$. The σ is thus ~ 10 and ~ 3 respectively. Thus, many nodes will have very few edges for $q = 0.01$, slowing down the propagation

The Time Dimension

- Up to now, only “space” in the form of topological population structures has entered into the picture
- Time has been considered *synchronous*; i.e., all the individuals act simultaneously at the ticks of a global clock
- But does this global synchronization make sense or is it only a useful abstraction?

Asynchronous Evolution

- Synchronous evolution is simple and can be used in artificial systems, where no physical limitation exists
- Asynchronous evolution is more complex but it is more faithful to Nature. No global clock. Signals can only travel at finite speed in physical and biological systems
- Since there can be many different sequential update orders for a cellular system, asynchronous evolution gives another degree of freedom to play with

Asynchronous Evolution: the Models

Three asynchronous evolution models will be used: *Line Sweep*, *Uniform Choice*, and *Random Sweep*

- In Line sweep (LS), the n cells are updated sequentially from left to right and line after line starting from the upper left corner cell.
- 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.

Asynchronous Evolution: the Models II

- 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 updating 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

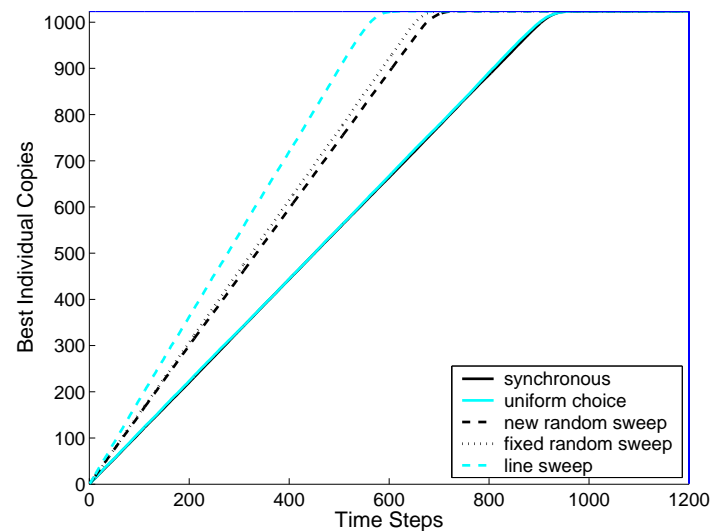
Asynchronous Evolution: Results

Results can be summarized as follows [8,9]:

- As in the synchronous case, asynchronous evolution in lattices produces a selection pressure that is lower than the panmictic case. The ranking does not change, with selection being more intense in mixing populations than in grids, which is in turn more intense than rings
- Selection intensity using asynchronous evolution is slightly stronger than for the synchronous case for the same topological parameters. Uniform choice is close to synchronous
- In a given topology, different asynchronous update methods give rise to different global induced selection pressures
- Thus, selection intensity in cellular populations can be changed, even *dynamically*, by using different cell update methods, different grid or neighborhood ratios, or both

Asynchronous Evolution: Results

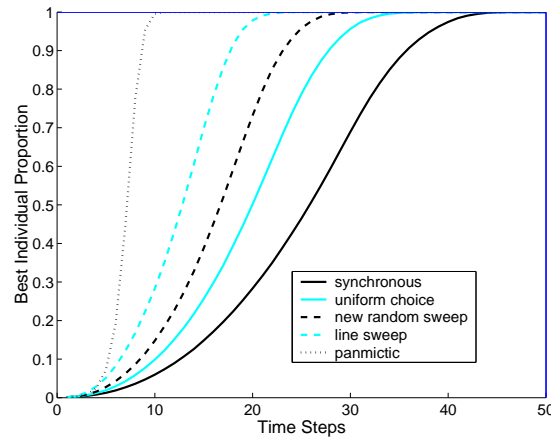
Takeover Times results for **rings** for various update methods



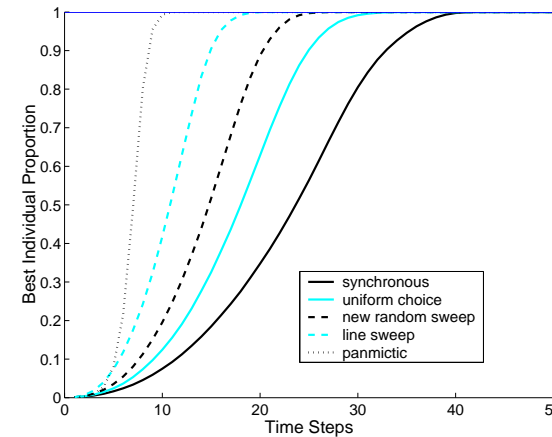
Takeover times with binary tournament selection: mean values over 100 runs. The vertical axis represents the number of copies of the best individual as a function of the time step

Asynchronous Evolution: Results

Takeover Times results for **tori** for various update methods



(a)



(b)

Takeover times with (a) binary tournament selection, and (b) linear ranking. Mean values over 100 runs. The vertical axis represents the number of copies $N(t)$ of the best individual in each population as a function of the time step t

What About “Real” Cellular EAs?

- Typical benchmarks have been used, both continuous and discrete
 - massively multimodal deceptive problems (MMDP)
 - satisfiability (SAT) problems
 - multimodal problem generator (P-PEAKS)
 - maximum cut of a graph (MAXCUT)
 - scheduling problems (MTTP)
 - continuous functions such as: Frequency Modulation Sounds (FMS), Ackley, Rastrigin etc.
- Those cover most classes of problems found in practice and should give an indication as to the observed tendencies

(The problems and the experiments are described in [4])

Parameters Used in the Cellular EA runs

<i>Population Size</i>	400 individuals
<i>Selection of Parents</i>	binary tournament + binary tournament
<i>Recombination</i>	DPX, $p_c = 1.0$
<i>Bit Mutation</i>	Bit-flip, $p_m = 1/L$ ($10/L$ for FMS)
<i>Individual Length</i>	L
<i>Replacement</i>	Rep_if_Better

Table 1: Parameterization used in the algorithm for the binary encoded problems. DPX indicates standard double point crossover.

Name	(shape of population)	Value of ratio
Square	(20 × 20 individuals)	0.11
Rectangular	(10 × 40 individuals)	0.075
Narrow	(4 × 100 individuals)	0.031

Table 2: Studied ratios.

Summary of Results

Algorithm	Avg. Solution (best=20)	Avg. Generations	Hit Rate
Square	19.813	214.2	57%
Rectangular	19.824	236.1	58%
Narrow	19.842	299.7	61%
LS	19.518	343.5	23%
FRS	19.601	209.9	31%
NRS	19.536	152.9	28%
UC	19.615	295.7	36%

Table 3: MMDP problem with a maximum of 1000 generations.

Algorithm	Avg. Solution (best=1)	Avg. Generations	Hit Rate
Square	1.0	51.8	100%
Rectangular	1.0	50.4	100%
Narrow	1.0	53.9	100%
LS	1.0	34.8	100%
FRS	1.0	38.4	100%
NRS	1.0	38.8	100%
UC	1.0	40.1	100%

Table 4: P-PEAKS problem with a maximum of 100 generations.

Algorithm	Avg. Solution (best \geq 100)	Avg. Generations	Hit Rate
Square	90.46	437.4	57%
Rectangular	85.78	404.3	61%
Narrow	80.76	610.9	63%
LS	81.44	353.4	58%
FRS	73.11	386.2	55%
NRS	76.21	401.5	56%
UC	83.56	405.2	57%

Table 5: FMS problem with a maximum of 3000 generations.

Algorithm	Avg. Solution (best=56.74)	Avg. Generations	Hit Rate
Square	56.74	11.3	100%
Rectangular	56.74	11.0	100%
Narrow	56.74	11.9	100%
LS	56.74	9.5	100%
FRS	56.74	9.7	100%
NRS	56.74	9.6	100%
UC	56.74	9.6	100%

Table 6: MAXCUT problem with a maximum of 100 generations.

Algorithm	Avg. Solution (best=0.02439)	Avg. Generations	Hit Rate
Square	0.02439	8.4	100%
Rectangular	0.02439	8.3	100%
Narrow	0.02439	8.9	100%
LS	0.02439	5.9	100%
FRS	0.02439	6.2	100%
NRS	0.02439	6.3	100%
UC	0.02439	6.3	100%

Table 7: MTTP problem with a maximum of 50 generations.

Algorithm	Avg. Solution (best=430.0)	Avg. Generations	Hit Rate
Square	429.54	703.1	79%
Rectangular	429.67	706.3	84%
Narrow	429.61	763.7	81%
LS	429.52	463.2	78%
FRS	429.67	497.7	85%
NRS	429.49	610.5	75%
UC	429.50	725.5	76%

Table 8: SAT problem with a maximum of 3000 generations.

Algorithm	Avg. Solution (best ≤ 0.1)	Avg. Generations	Hit Rate
Square	0.0999	321.7	78%
Rectangular	0.0994	293.1	73%
Narrow	0.1037	271.9	65%
LS	0.0932	302.0	84%
FRS	0.0935	350.6	92%
NRS	0.0956	335.5	87%
UC	0.0968	335.0	85%

Table 9: ACKL problem with a maximum of 500 generations.

Algorithm	Avg. Solution (best ≤ 0.1)	Avg. Generations	Hit Rate
Square	0.0900	323.8	100%
Rectangular	0.0883	309.8	100%
Narrow	0.0855	354.2	100%
LS	0.0899	280.9	100%
FRS	0.0900	289.6	100%
NRS	0.0906	292.2	100%
UC	0.0892	292.4	100%

Table 10: RASTR problem with a maximum of 700 generations.

On the whole, results agree with expected selection pressures

Island Models

- Island models have been often used in Evolutionary Computation
- The most complete modeling and analysis has been done by Cantú-Paz for GAs [3]
- Empirically, they have been found nearly always more efficient than the panmictic population model
- Here we will focus on Multipopulation Genetic Programming

Multi-Population Genetic Programming

A number of parameters must be considered:

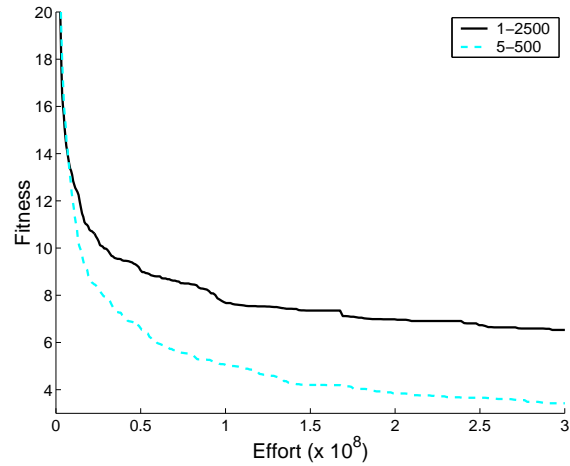
- The number of islands (subpopulations)
- The size of the subpopulations
- The communication topology
- The number and type of migrating individuals
- The frequency of migration

These parameters have been empirically investigated in Fernández et al.[5], on standard and real-life problems. Details of the test problems and results can be found there

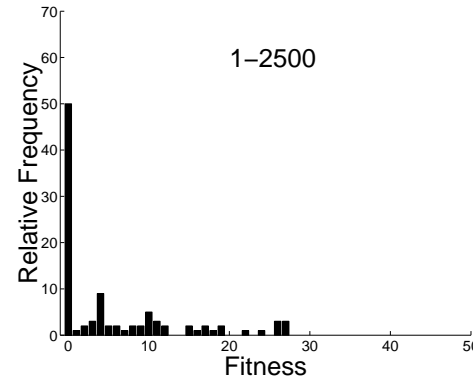
Multi-Population Genetic Programming: Results

- In general, multi-population GP is more efficient than standard panmictic GP on those problems: **better results with the same computational effort**
- For a given total population size, there is a preferred interval for subpopulation size which is problem-dependent
- If the subpopulations are too small, island GP does not perform well
- The “optimal” number of individuals to exchange is about 10% of the subpopulation size; the frequency of exchange should be between 5 to 10 generations independent of the problem
- The influence of inter-island communication topology is comparatively less important

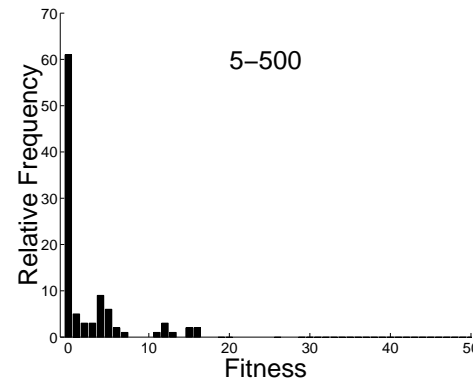
Experimental Results: Ant Problem



average effort



relative frequency of solutions panmictic GP

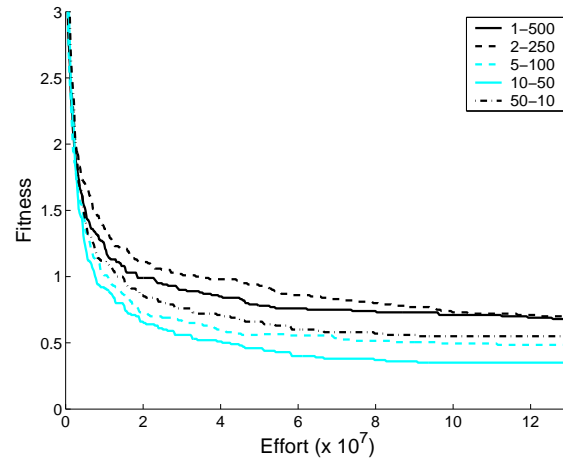


relative frequency of solutions multipop GP

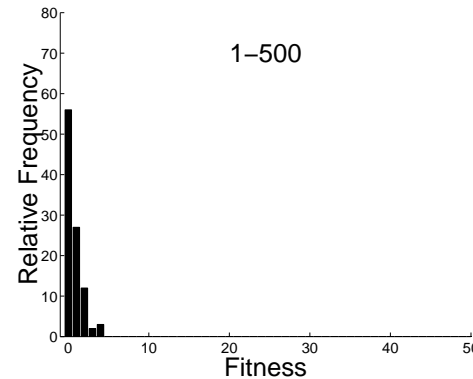
	$E = 1 \times 10^8$	$E = 2 \times 10^8$	$E = 3 \times 10^8$
1- 2500	43	47	50 ($\sigma = 5.000$)
5- 500	53	60	61 ($\sigma = 4.877$)

hit rate/100 runs

Experimental Results: Even Parity Four



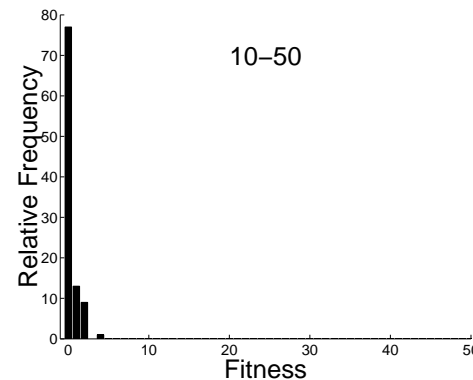
average effort



relative frequency of solutions panmictic GP

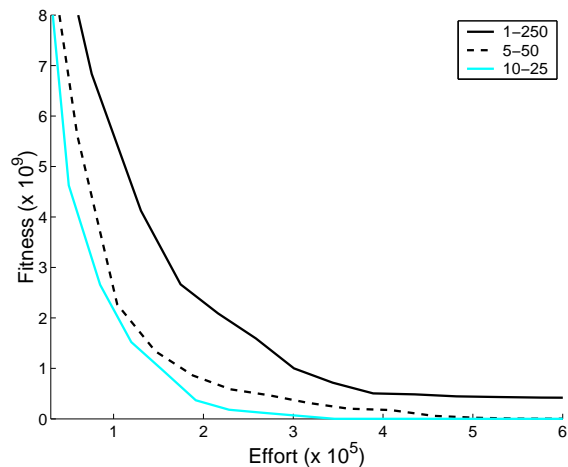
	$E = 8 \times 10^7$	$E = 10 \times 10^7$	$E = 12 \times 10^7$
1- 500	53	54	56 ($\sigma = 4.964$)
2- 250	55	60	60 ($\sigma = 4.899$)
5- 100	64	65	65 ($\sigma = 4.770$)
10- 50	76	77	77 ($\sigma = 4.208$)
50- 10	62	64	65 ($\sigma = 4.770$)

hit rate/100 runs

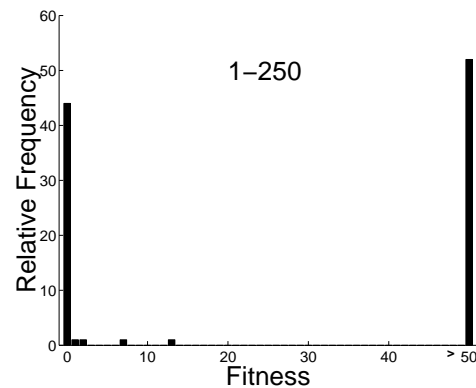


relative frequency of solutions multipop GP

Experimental Results: Symbolic Regression



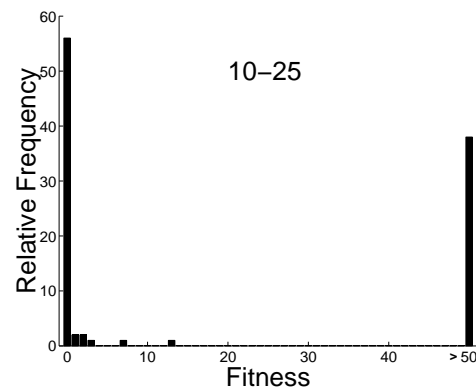
average effort



relative frequency of solutions panmictic GP

	$E = 2 \times 10^5$	$E = 3 \times 10^5$	$E = 4 \times 10^5$
1- 250	38	44	44 ($\sigma = 4.964$)
5- 50	51	54	55 ($\sigma = 4.974$)
10- 25	53	55	56 ($\sigma = 4.964$)

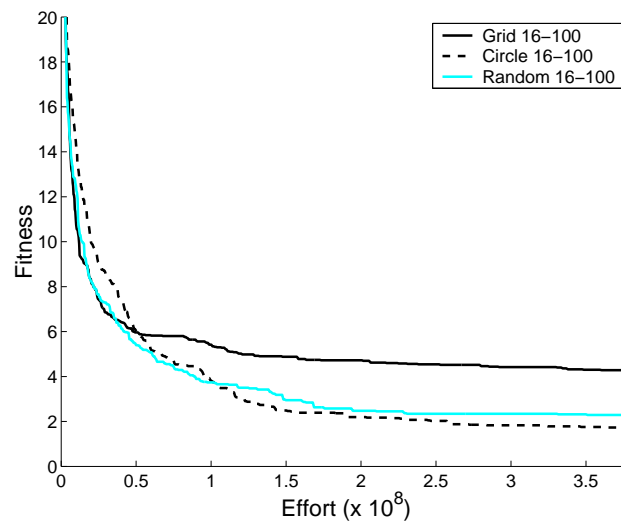
hit rate/100 runs



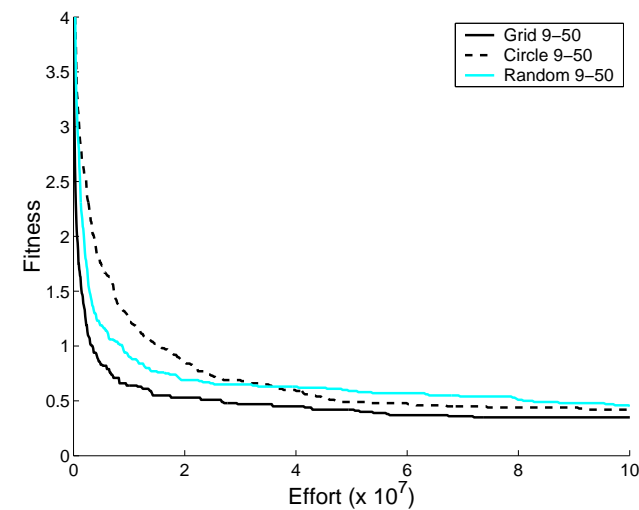
relative frequency of solutions multipop GP

Comparing Topologies

The empirical result is that, for island models, the precise migration topology is relatively unimportant, at least for the cases studied here. This is reasonable, given that evolution is still mainly panmictic



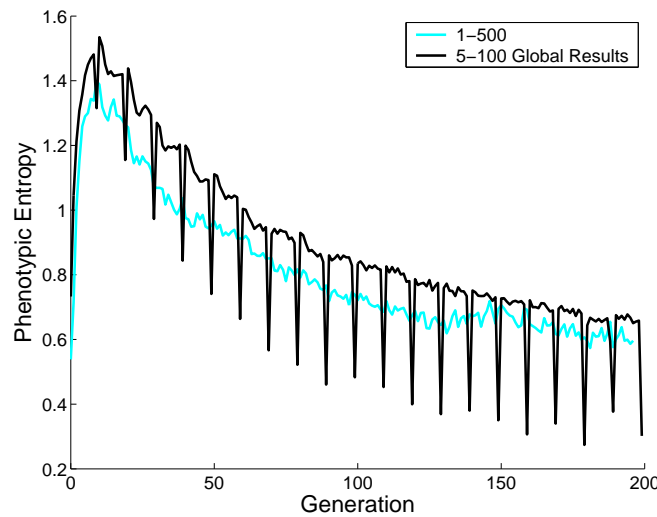
Ant problem



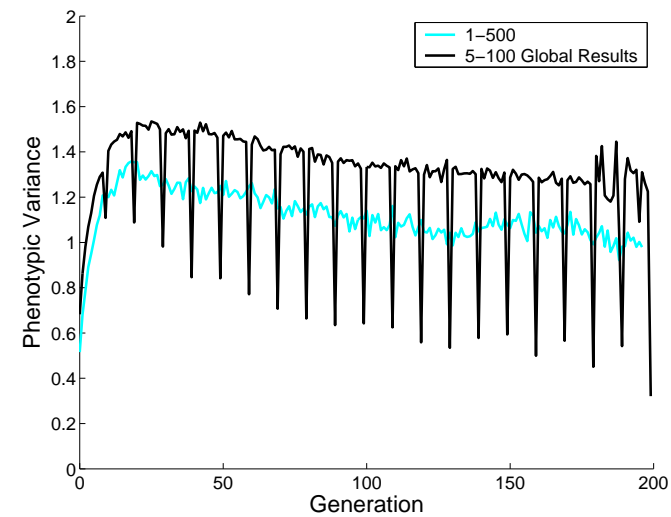
Even Parity 4 problem

Maintaining Diversity in Island GP

A better global phenotypic diversity during the run seems to be correlated with the good results obtained with multi-population GP

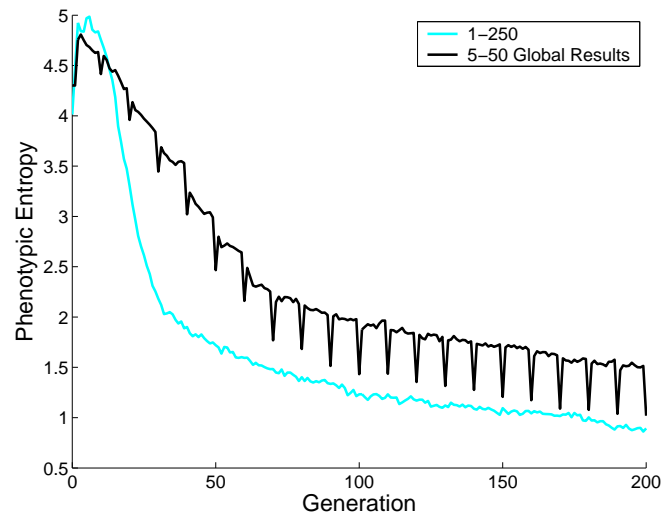


Even Parity - Entropy

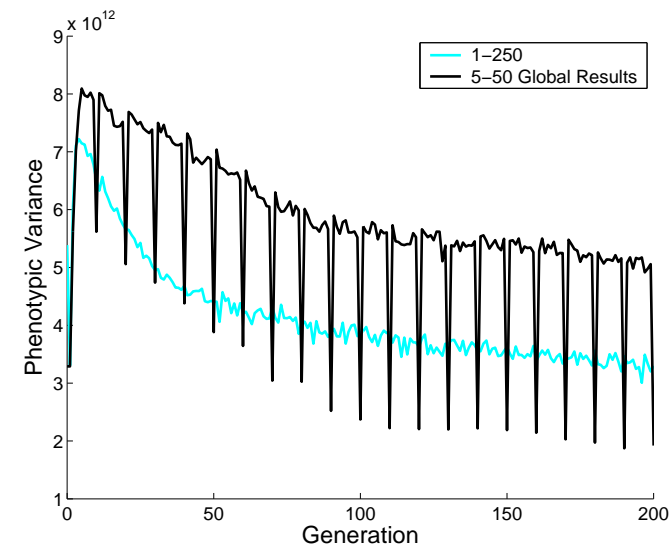


Even Parity - Variance

Maintaining Diversity in Island GP



Symbolic Regression - Entropy



Symbolic Regression - Variance

Effectivity of Multi-Population EAs

Summarizing, and extending to other island EAs for which many results exist:

- Most empirical results tend to show that island EAs are more efficient than panmictic EAs
- The effectivity of multi-population EAs seems to depend on the nature of the problem
- Overall population diversity is better maintained in a multi-population setting
- Separable problems and problems with multiple solution paths seem to be more suitable for the distributed approach

A Note on Implementation I

- We have been talking of *models*, without any implementation details
- All the models described previously can be implemented as sequential algorithms on sequential architectures
- However, they are easy to implement on parallel or distributed architectures with good **performance gains**
- This is because communication and synchronization overheads are minimal for EAs (except for asynchronous cellular EAs)

A Note on Implementation II

- Island models can be very easily and efficiently implemented on dedicated clusters (Beowulf-style systems), with both synchronous and asynchronous migration patterns, using message-passing libraries (e.g. MPI)
- Synchronous cellular systems can be implemented on clusters by using domain decomposition techniques and message passing for the domain borders [7]
- Load balancing is only needed for cellular GP systems [6]

To Know More

- [1] E. Alba and M. Tomassini. *Parallelism and evolutionary algorithms*. IEEE Transactions on Evolutionary Computation, 6(5):443–462, October 2002.
- [2] D. Andre and J. R. Koza. *Parallel genetic programming: A scalable implementation using the transputer network architecture*. In P. Angeline and K. Kinnear, editors, *Advances in Genetic Programming 2*, pages 317–337, Cambridge, MA, 1996. The MIT Press.
- [3] E. Cantú-Paz. *Efficient and Accurate Parallel Genetic Algorithms*. Kluwer Academic Publishers, Boston, 2000.
- [4] B. Dorronsoro, A. Alba, M. Giacobini, and M. Tomassini. *The Influence of Grid Shape and Asynchronicity on Cellular Evolutionary Algorithms*. In *Proceedings of CEC '04*, IEEE Press, 2004
- [5] F. Fernández, M. Tomassini, and L. Vanneschi. *An empirical study of multipopulation genetic programming*. *Genetic Programming and Evolvable Machines*, 4:21–51, 2003.
- [6] G. Folino, C. Pizzuti, and G. Spezzano. *A cellular implementation of parallel genetic programming*. IEEE Transactions on Evolutionary Computation, 7(1):37–53, 2003. October 2002.
- [7] G. Folino, C. Pizzuti, G. Spezzano, L. Vanneschi, and M. Tomassini. *Diversity analysis in cellular and multipopulation genetic programming*. In *Congress on Evolutionary Computation (CEC'03)*, 305–311, Canberra, Australia. IEEE Press.

- [8] M. Giacobini, M. Tomassini, and A. Tettamanzi. Modelling selection intensity for linear cellular evolutionary algorithms. In P. Liardet et al., editor, *Proceedings of the Sixth International Conference on Artificial Evolution, Evolution Artificielle 2003*. Springer Verlag, Berlin, 2003. To appear.
- [9] M. Giacobini, E. Alba, A. Tettamanzi, and M. Tomassini. *Modeling selection intensity for toroidal cellular evolutionary algorithms*. In Proceedings of GECCO'04, Springer Verlag, Berlin, 2004.
- [10] B. Manderick and P. Spiessens. *Fine-grained parallel genetic algorithms*. In J. D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, pages 428–433. Morgan Kaufmann, 1989.
- [11] J. Sarma and K. A. De Jong. *An analysis of the effect of the neighborhood size and shape on local selection algorithms*. In H. M. Voigt et al. editors, *Parallel Problem Solving from Nature (PPSN IV)*, Lecture Notes in Computer Science vol. 1141, pages 236–244. Springer-Verlag, Berlin, 1996.
- [12] J. Sarma and K. A. De Jong. *An analysis of local selection algorithms in a spatially structured evolutionary algorithm*. In T. Bäck, editor, *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 181–186. Morgan Kaufmann, 1997.
- [13] D. Whitley, S. Rana, and R. B. Heckendorn. *Island model genetic algorithms and linearly separable problems*. In D. Corne and J. L. Shapiro, editors, *Evolutionary Computing: Proceedings of the AISB Workshop*, Lecture notes in computer science, vol. 1305, pages 109–125. Springer-Verlag, Berlin, 1997.