

Chapter 3

MANIPULATION OF CONVERGENCE IN EVOLUTIONARY SYSTEMS

Gearoid Murphy¹ and Conor Ryan¹

¹*Biocomputing and Developmental Systems Group, University of Limerick, Ireland*

Abstract Convergence is a necessary part of any successful GP run, but is also a great weakness due to material lost through the hemorrhage of genetic content through the established evolutionary dynamics. This chapter examines the phenomenon of convergence commonly observed in evolutionary systems before introducing a number of functionally distinct mechanisms are analysed and tested with respect to their ability to modulate the loss of potentially useful genetic content. Each of these methods use little or no explicit measurements to calculate diversity, and we show that they can have a dramatic effect on empirical performance of GP.

Keywords: convergence manipulation

1. Introduction

Convergence in any domain intuitively invokes analogies of stabilisation, specialisation and exploitation. In the domain of genetic programming, convergence refers to the characteristic behaviour of the population dynamics; wherein the initial random set of expressions undergo a series of alterations which results in an increase in the relevance of the expressions to the requirements of the target domain. The concentration of the content of the expressions within a relatively constrained area of the expression space is another effect of convergence.

In general the term of convergence is used fairly intuitively within the Genetic Programming community and is usually meant to invoke the previous description. For the duration of the chapter we will also subscribe to this generic philosophy, in favour of more specific definitions.

The notion of convergence has particularly broad connotations within the context of AI paradigms, more specifically within that subset of AI techniques

which utilise some stochastic process in their initial projections. This is self-evident in GP wherein the systems initial random projection is convoluted by the evolutionary dynamics so as to conform to the objective feedback signal but is also observed in gradient descent methods such as Neural Networks.

In the context of Neural Networks, the phenomenon of convergence manifests itself as the system's ever increasing conformity with the training signal. The effectiveness of Neural Networks as a search method and for gradient descent methods in general depends heavily on what is referred to as the alpha factor. The parameter modulates the magnitude of the changes in the "position" of each learning weight with respect to the output error. Choosing an alpha factor which is too large means that the "jumps" around the expression space are too large, and potentially useful states are missed, alternatively choosing an alpha factor which is too small means that a great deal of unproductive areas of the expression space are evaluated.

This effect of the alpha factor on the convergence behaviour of neural networks succinctly summarise the challenges associated with exploiting the dynamics of convergence in the need to adequately explore the expression space whilst at the same time exploiting the areas of the expression space already occupied. Unfortunately, no single parameter can account for the convergence behaviour of genetic algorithms in so salient a manner as the alpha parameter does for Neural Networks, and we are instead faced with number of possibly interacting parameters, such as population size and selection methods.

The primary consequence of this observation is the motivation to investigate, understand and if possible exploit the dynamics of convergence within a Genetic Programming context. This will improve the viability of the paradigm as an AI solution. To achieve this we put forward a series of experiments whose purpose is to show the potential of the experimental techniques to improve the genetic content of the population whilst as the same time preventing the loss of potentially useful sub solutions.

The layout of the paper is as follows, Section 2 gives some background on the previous work done on the manipulation of convergence. Section 3 describes the specific parameters that are common to all experiments. The experimental work then follows. A discussion of the work and future aspirations is given in Section 8 followed by our conclusions in Section 9.

2. Background

Ever since the inception of contemporary evolutionary algorithms in Hollands seminal work, (Holland, 1975), the proliferation of a single dominant solution throughout the population and the consequent evolutionary gridlock it results in, has been recognised as a core concern of the science. This sec-

tion reviews some of the work done to address this issue and the nature of the resulting strategies produced.

Using the fitness of the population as a mechanism to prevent premature convergence was one of the first such strategies developed in; (Goldberg and Richardson, 1987). By modulating the fitnesses of the population to inhibit dense congregations of homogeneous solutions, *fitness sharing* techniques attempted to segregate the population into *niches*. Inspired by observations of niche specialisation within natural biological ecosystems, many variants of this strategy have been pursued such as sequential niching, (D. Beasley and Martin, 1993), speciation with implicit fitness sharing and co-evolution, (Darwen and Yao, 1997) and a niching method known as clearing, (Sareni and Krahenbuhl, 1998).

While effective at encouraging the occupation of multiple local optimum, fitness sharing methods suffer from the need to set a priori parameters such as the similarity measure needed to define a minimum distance between optima. Such measures are hard to define and may need to change in the later stages of evolution, (Sareni and Krahenbuhl, 1998). Using fitness as the primary segregation measure has also been demonstrated in the Hierarchical Fair Competition Model, (Hu and Goodman, 2002).

Using age as the method by which convergence may be inhibited has also been investigated. The Age Layered Population Structure (ALPS), (Hornby, 2005) has been used to great effect on a difficult optimisation problem. The method uses age “bands” to define the population pools the individuals of the population may occupy. New individuals are continually introduced so as to maintain exploration. Normal evolutionary dynamics are present within the age “bands”.

An alternative way in which age may be used to modulate population convergence is given in (Naoyuki Kubota and Shimojima, 1994). This technique undertakes the hypothesis that the effectiveness of convergence is inhibited by the loss of potentially useful genetic content due to the replacement strategies of steady state wherein the worst individual is replaced by the new offspring. This dynamic is inhibited by allowing individuals to exist for multiple evolutionary cycles within the population. An age value is used to determine how many cycles they may exist on the population. So, rather than trying to *segregate* the population, this method attempts to *protect* relatively poor individuals, thus allowing them the opportunity to propagate their genetic content.

Both age and fitness have thus far been demonstrated as valid strategies for manipulating the dynamics of convergence. Another strategy discussed here employs spatial segregation and is known as the *Island Model*, (Dehmeshki et al., 2003). In this model, multiple initial populations are allowed to evolve, the idea being that each population will develop towards different optima. After a pre-specified number of evolutionary cycles, the “islands” are allowed to in-

tercommunicate. By such communication, the models are producing a number of beneficial effects.

Primarily, the homogenisation of each island is delayed with the introduction of such new material, with beneficial consequences for the ultimate convergence of the entire population of islands. Also, separate aspects of the final solution may be present in different islands, by communication these sub solutions are allowed to coexist and intermingle. The incompatibility of highly evolved solutions of the same domain but from different evolutionary runs is a well known phenomenon, and inter island communication avoids this by maintaining coherence between the islands.

An alternative mechanism for preventing convergence is to inhibit crossover events between individuals whose genetic lineage is similar, with the intention of preventing the proliferation of a dominant “family” of expressions, (Braught, 2005). This mechanism also encourages exploration by forcing solutions whose origin lies in relatively different areas of the expression space into crossover events.

This concludes our background for the work on manipulating convergence. The domain is so vast that an exhaustive description of all the techniques is infeasible but we feel we have described the work with most relevance to the experiments described in this chapter.

3. Overview of experiments

The parity domain is presented as the domain of choice for the experiments performed in this chapter. This problem is difficult without the inclusion of the XOR or EQ functions, thus providing a reasonably difficult test of the viability of the experiments whilst at the same time being reasonably manageable in terms of time to compute. All parity experiments used 5 inputs with a maximum attainable fitness is 32. Each experimental result is the averaged behaviour of 30 independent runs.

The experimental parameters used to define the runs are given in Table 3-1. These remain constant for all experiments except where explicitly stated otherwise. The function set used by the experiments is given in Table 3-2. All populations were initialised using the ramped half and half technique taken from (Koza, 1994). All models use the steady state evolutionary dynamics unless explicitly stated otherwise. Individual sections of this chapter contain any further details needed to implement the experiments described in them.

The order of the experiments reflect the chronology of their execution. Each experiment begins with a particular hypothesis and tests it in some way. Discrepancies between the projected behaviour and the observed dynamics serve as focal points for the development of further hypotheses.

Table 3-1. Experimental parameters used throughout the chapter. Deviation from the parameters will be explicitly stated during each experimental instance.

Parameter	Value
pop size	100
tournament size	2
max initial depth	6
min initial depth	4
mutation rate	0.0

Table 3-2. Functional primitives used in all experiments.

Function	Arity
And	2
Or	2
Nand	2
Nor	2

In order to provide insight into the dynamics of the behaviour of the models, we employ fitness comparison graphs which plot the average best fitness characteristics of the systems against a standard GP system. Error bars are omitted from the fitness graphs for clarity. We will also exploit the fact that our experiments are situated in a Boolean domain by mapping the binary output sequence associated with each expression over the entire set of test cases onto an integer number. By counting the number of unique integers within the population of candidate solutions we can derive a sense of how well the various strategies are maintaining different independent solutions. These simple representations of solution diversity give an immediate intuition about the convergence behaviour of the systems.

Explicit metrics of phenotypic structural diversity are not shown as they were never used in the actual experimental processes and to include them out of context would detract from the primary motivation of the paper, which is to manipulate convergence, not to promote such diversity. Improving convergence and promoting such diversity are not necessarily mutually exclusive but convergence can be improved with a loss of such diversity metrics.

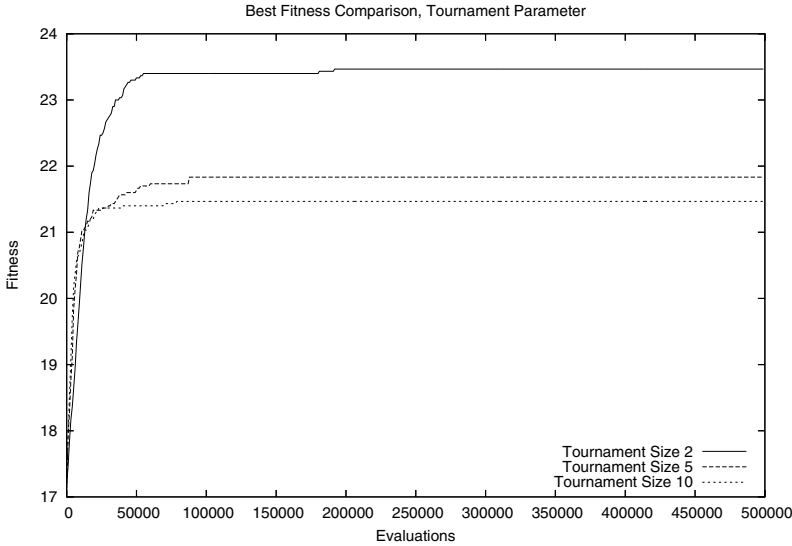


Figure 3-1. Best fitness averaged over 30 independent runs for a variety of tournament sizes. A tournament size of two randomly selects parents with the fitter parent contributing the root branch in the crossover event.

4. Experiment : Modulation of Tournament Size and Population Size

This section presents a number of experiments showing the effect that modulating a number of different parameters and mechanisms can have on the dynamics of convergence. We stress that these results are well established within the community and that conventional wisdom would dictate that there is nothing to be gained from reinventing the wheel, however the conclusions drawn from these results are instrumental in the conception of the next experiment, so for the convenience of contextual congruity and effective communication we include them.

The first set of results in Figure 3-1 illustrates the effect of the size of the tournament on the population dynamics. It is immediately obvious that reducing the size of the tournament used to select the parents of a crossover event has a beneficial effect for the convergence dynamics of the population. In a situation where there is a relatively large tournament size, a few of the fittest individuals will consistently transmit the genetic content of their expressions amongst the population very quickly.

In rare situations, such high convergence rates will settle on a sustainable convergence dynamic wherein the quality of the population as it converges increases substantially. Unfortunately, such behaviour is the exception rather

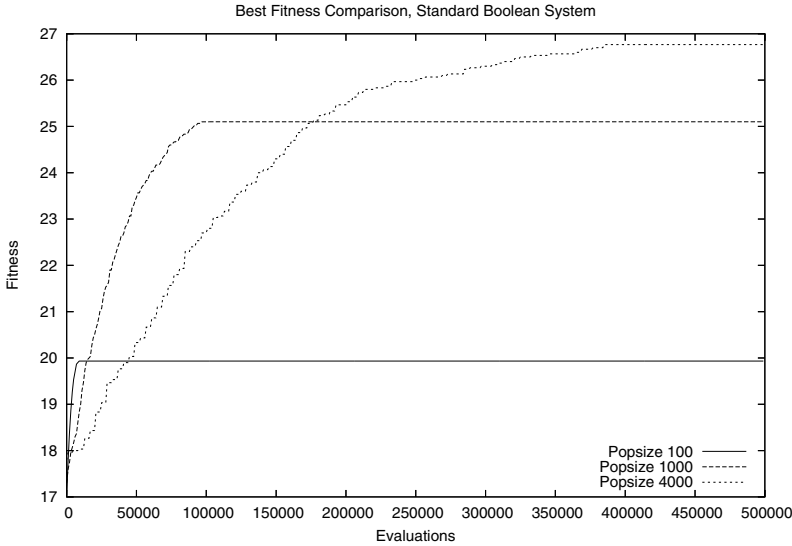


Figure 3-2. Best fitness averaged over 30 independent runs for a variety of population sizes. Increasing the population size delays premature convergence but does not avoid it.

than the rule, as high tournament sizes generally have a detrimental effect on the convergence behaviour of the system.

Our results showed that the best performing tournament strategy simply picked two parents at random, with the better parent contributing the root branch to the crossover event. If the resulting offspring was better than the worst parent, it was reinserted into the worst parents place.

The importance of this result is that it illustrates that the established dynamic of primarily proliferating the genetic content of the fittest individuals serves only to prematurely accelerate the system into an inescapable local optimum, particularly in a difficult domain such as parity.

The next set of results focused on the effect of changing the population size. Figure 3-2 shows the progression of the best fitness for population sizes 100, 1000 and 4000. It is well known that in order to achieve better results in a given domain using Genetic Programming, one can improve results simply by increasing the population size. This significantly increases the length of time it takes for one good individual to start dominating the population, thus allowing time for the exploration of less fit but potentially useful expressions, however even the largest population still converged before coming close to the max fitness of 32. However, simply increasing the population size, whilst undeniably effective, suffers from a harsh law of diminishing returns, and so is only effective up to a certain point. We use *solution density* graphs to demonstrate how the

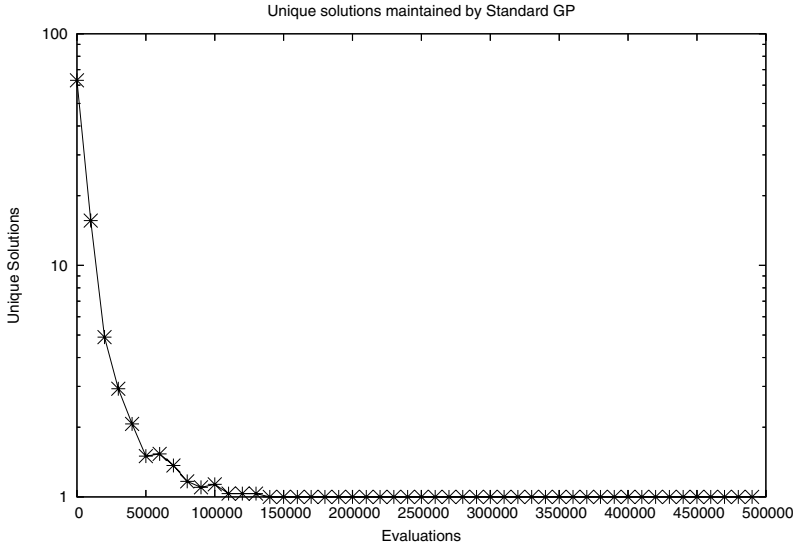


Figure 3-3. Diagram illustrating rapid loss of unique solutions within the population over time. Note that this measure does not perform any explicit diversity analysis of the genetic content of the population, rather, it looks at the environmental identity of the expressions, in the context of the boolean domains.

number of unique solutions changes amongst the population during the run. Consider the population in Figure 3-3.

The most pressing observation from these diagrams is that the pressure on the system to converge, even with the low tournament size, is very high. So high in fact, that the population quickly concentrates itself into dense pools of homogeneous expressions, exhibiting the characteristics of a “greedy” algorithm and consequently devastating its potential for improvement through parts of the expression space which do not immediately provide a return in fitness.

We conclude that the most important factor modulating the concentration of the population into dense pools of homogeneous expressions was not the prevalence of a single solution but the loss of so many potentially useful untried expressions. This motivated the next experimental setup wherein the loss of expressions was manipulated by using age-based dynamics.

5. Experiment : Dynamically Sized Population with Age Based Dynamics

This section describes an experiment implemented to test the hypothesis that by slowing the flow of relatively unfit but potentially useful genetic content out of the population, that the quality of the evolved population will be improved.

Table 3-3. Parameters used to implement the Age based system.

Parameter	Value
Population Size	100
Tournament Size	2
Population Increase	10
Positive Crossover Feedback	2
Negative Crossover Feedback	-1
Initial Individual Age	40

We implement this dynamic by changing one of the fundamental mechanisms in the tournament selection, wherein the worst individual of a tournament is *not* replaced by better offspring from the winning parents of the tournament. Instead each newly created individual is given a number of cycles of existence within the population. Furthermore we augment this dynamic by rewarding parents which successfully produce a valid offspring with more cycles of existence and punishing parents producing unfit offspring by reducing the number of evolutionary cycles they have been allocated.

After each generation, the produced offspring are sorted and a pre designated number of the best offspring are allowed into the population, controlled by the population increase parameter in Table 3-3. This combination of parameters results in a dynamically sized population. After the initial growth spurt, the population will stabilise at some size, depending on the chosen parameters, and stay around that point, as the flow of individuals out of the population matches the flow of new individuals into the population.

From Figure 3-4 we can see that a population using this method to does better than a population utilising a standard steady state algorithm with tournament selection and with an initial population ten times larger. This reinforces the hypothesis that manipulating the convergence dynamics can substantially improve the effectiveness with which a given expression space can be searched. Furthermore, it also shows us that even relatively small populations, will to a large extent, contain quite a lot of the expressions needed to solve the problem.

However, the approach is inconsistent, sometimes doing very well and sometimes not, Figure 3-5. Observing the fitness density graph in Figure 3-6 indicates why. In contrast to previously shown solution density graph taken from standard GP implementations, the age based system is clearly maintaining discrete pools of expressions. These pools can only be maintained by their being continually replenished with positive feedback from successful crossover events. This accounts for the systems resistance to convergence as evidenced by the improved

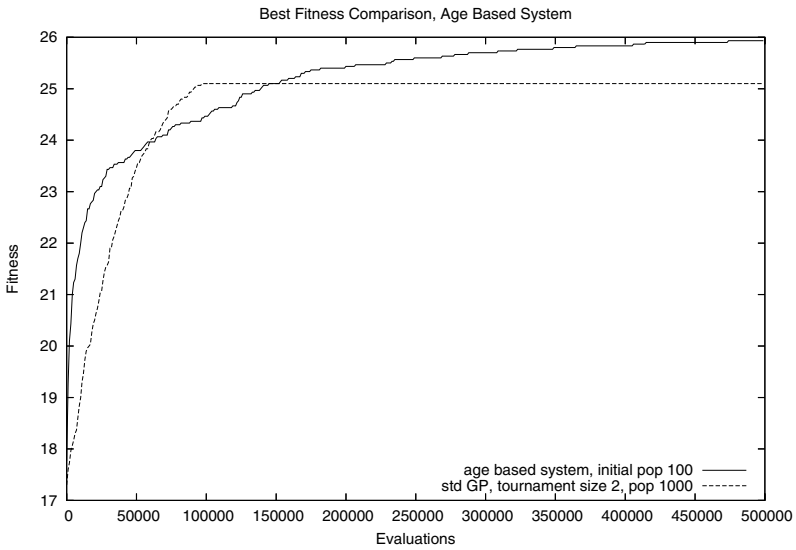


Figure 3-4. Best fitness averaged over 30 independent runs, comparison between age based model with standard GP model. Even though the standard system starts off with a population 10 times larger than the age model, it is still outperformed by the age model.

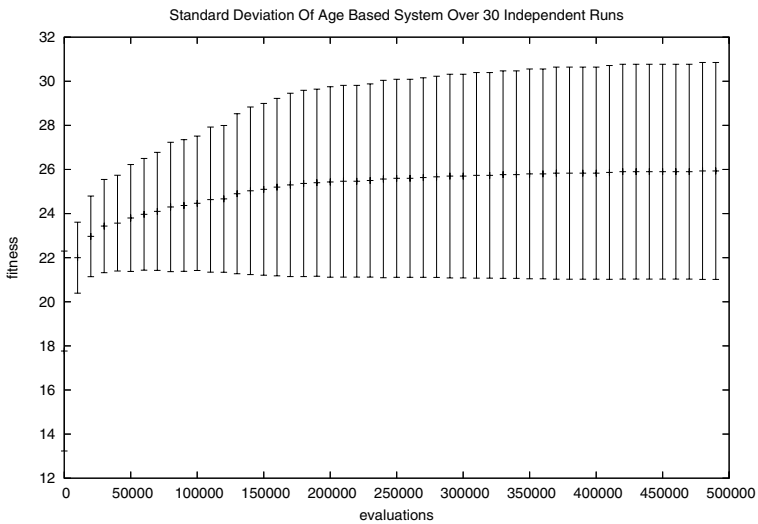


Figure 3-5. Figure showing high levels of statistical variance in performance of the age based dynamics model.

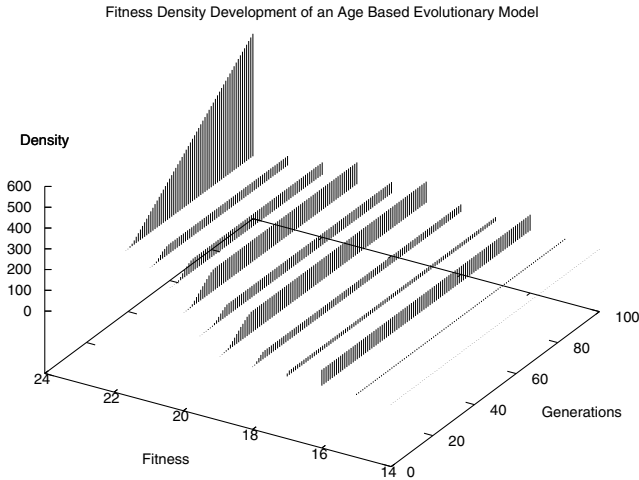


Figure 3-6. Fitness density diagram for the age model. Note the distinctly identifiable evolutionary “niches” occupied. These are maintained by the feedback from positive crossover events. Also, observe the increase in size of the highest fitness gradient, consistent with the dynamic population size of the model

results however, it is plainly evident that the vast majority of the population is still concentrating itself into a high density pool of expressions.

So, while the age-based dynamics have improved the performance to some extent, the system is still vulnerable to situations where the pressure to converge onto an sub optimum is too powerful to resist and consequently neutralises the benefit of the age based dynamics.

Given the observation that convergence is such an unavoidable evolutionary phenomenon, how can we improve its chances of avoiding sub optimal attractors?

6. Experiment : Interacting Subpopulations

Genetic Programming implementations such as RTL, (Keijzer et al., 2005), have demonstrated the viability of improving the convergence behaviour of evolutionary systems by continually improving the functional content of the populations initial generation. However these are most effective in domains where the functional structure is relatively salient and are not as generally applicable to a broad range of problems like the methods presented here.

This experiment attempts to mimic aspects of RTL’s behaviour in that it attempts to continually improve the initial genetic content for a sequence of evolutionary runs as well as implementing the necessary functionality to pre-

vent the dense homogeneous expression pools characterising highly converged populations as observed in Section 4.

Our design solution to these requirements was an evolutionary model consisting of a population pool which is generated from the expressions derived from *multiple evolutionary* runs of subpopulations of an earlier population pool. The subpopulations are obviously much smaller than the population pool – in fact in our runs we have found that the best size for a sub population is only two individuals. Given that we can expect convergence to produce a highly homogeneous subpopulation at the end of the run, having a small subpopulation size prevents the propagation of a particularly dominant individual.

The subpopulations are seeded with random individuals from the current population pool, ignoring any preference applicable due to fitness, so as to improve the convergence dynamics in the manner observed in the experiments in Section 4.

The subpopulations are allowed to run for a specified number of generations. At the end of a subpopulation run, a taboo filter is passed over the converged subpopulation so as to prevent duplicate expressions entering into the next generation population pool. The taboo filter is a simple test for equality between two expressions. As the individuals are picked at random and as a considerable number of subpopulation runs may be needed to generate the next population, it is highly likely that a single individual may be part of multiple subpopulations.

Figure 3-7 shows the best fitness performance of the system for a number of different subpopulation generation lengths. This parameter was considered the most influential on the behaviour of the system as it directly controlled the investment of effort made by the system in improving the genetic content of the subpopulations. This can be seen quite clearly in the graph as the higher subpopulation generations exhibit the best performance.

The relationship between the interacting subpopulation model and previously established evolutionary models should be acknowledged. In particular the similarities between Island Models and the interacting subpopulation model are strong. The subpopulation model was originally envisaged as an iterative mechanism to improve the content of a population whilst slowing convergence. Island models provide the same form of functionality, albeit in a different method of execution, what it illustrates though is the validity of mixing the results of multiple sub evolutions so as to provide a satisfactory strategy of both exploitation and exploration. Also, when the subpopulations are only allowed to evolve for 1 generation, the system is very similar to a generational model.

As an instantiation of the hypothesis that improving the content of a population whilst preventing the homogenisation of the expressions in the population, the interacting subpopulation model is reasonably successful. One of the core principles of the system, selecting the subpopulations at random, was imple-

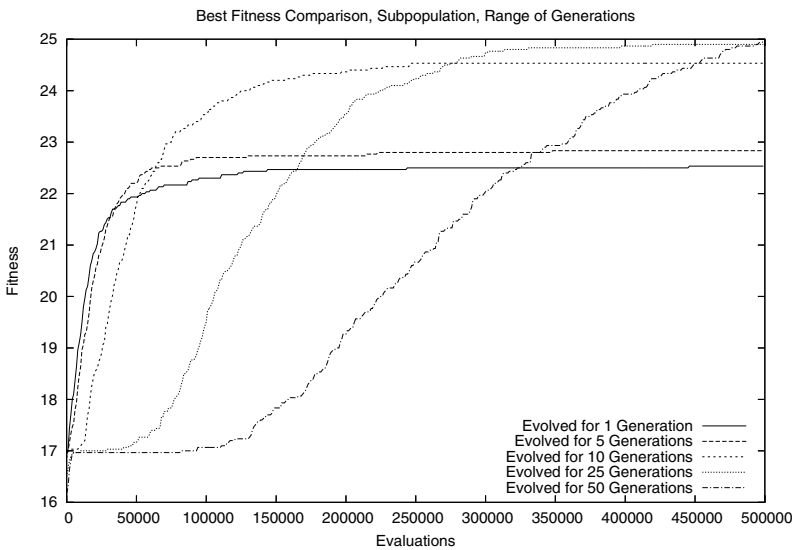


Figure 3-7. Best fitness averaged over 30 independent runs. Each experiment modulated the amount of time each subpopulation spent evolving. Its effect is clearly beneficial to the system, however it can inhibit the performance increase of the population when too much time is spent attempting to improve the fitness of the subpopulations, as evidenced by the experiment which used 50 generations to improve the subpopulations. The latency observed in the fitness of the higher generation experiments is symptomatic of the high number of evaluations applied to the very poor initial content of the populations.

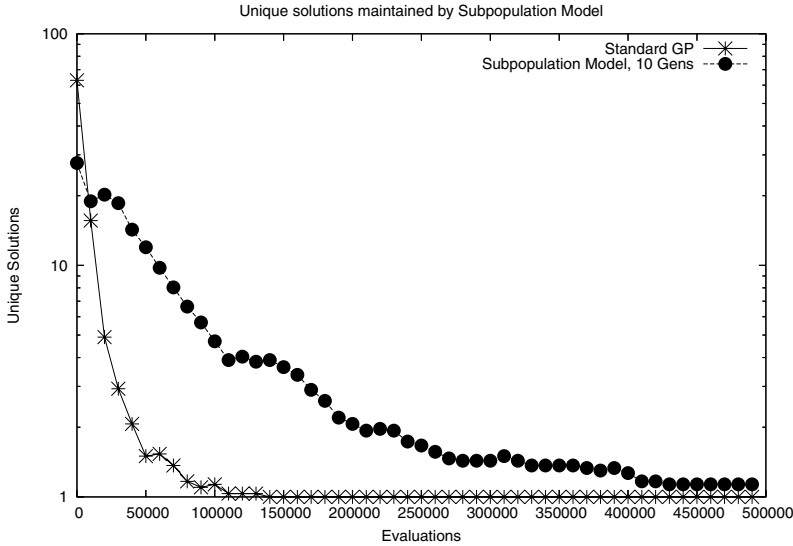


Figure 3-8. Solution density development for an interacting sub population model. Note the systems relative resistance to being trapped by a single sub-optimum, however, it is still susceptible to such inescapable attractors

mented on the observation from Section 4 that by relieving the selection pressure one could prevent the rapid takeover of a population by a particularly dominant expression.

However, this practice does not completely eliminate the phenomenon of dominance by a sub-optimal individual, as the evolutionary dynamics do not facilitate the mixing of expressions from relatively different areas of the search space. The next experiment attempts to do just that.

7. Experiment : Hereditary Repulsion

This section details the final experiment presented for this chapter. It was designed to offset the homogenisation of a population of expressions by forcing different areas of the expression space to mix. By continually facilitating a vigorous convolution of the genetic content as well as only accepting individuals with a marked improvement over their parents, this evolutionary model attempts to capture the best aspects of both exploration and exploitation.

The most distinctive aspect of its functionality is the manner in which it encourages different location areas in the expression space to be involved in the crossover events. It does this through hereditary repulsion. The idea behind the mechanism is that each expression has a specific family tree associated with it. By taking two individuals and counting the number of common ancestors

they have, a rough measure of their common lineage can be derived. The notion of lineage as measure to increase diversity by seeding tournaments with diverse lineages has been examined in (Braught, 2005). This setup can still allow relatively high fitness individuals to start dominating the population. Our modification to the lineage paradigm is to base the tournaments on minimising the hereditary overlap.

To this effect, our selection method picks an expression at random. It then picks a tournament set of expressions and determines the hereditary overlap between the initial random expressions and the tournament set of expressions. The expression with the least overlap is judged as being the most viable one to combine. In this way, exploration is encouraged.

The next critical aspect of the model's behaviour is that the resulting expression from the crossover must be better than both its parents. This mechanism is in place to ensure that the population content will always improve. A taboo operator is employed to prevent the proliferation of a single good expression. The model operates on a generational population mechanism.

It is evident to the informed reader that such a mechanism will result in a significant number of discarded expressions. However, as shown in Figure 3-9, the model produces excellent results despite this, outperforming all previous implementations. These results are a strong confirmation of the hypothesis that; by providing a powerful combination of both exploration and exploitation; the convergence dynamics of the population of expressions will be significantly improved. The ability of Hereditary Repulsion to avoid convergence is clearly seen in the solution density Figure 3-10.

8. Discussion and Future Work

This section presents a discussion on the papers contents and future work.

Utilisation of the Models

All the models presented in this paper are experimental systems. They were not meant to provide an exhaustive analysis of the various operating characteristics of the algorithms. Because of this, there most likely exist a significant number of augmentations to these systems which could further improve their performance. Work done in (Hornby, 2005), for example, illustrates the effectiveness of a well executed age based dynamics evolutionary system.

The effectiveness of the hereditary repulsion model is an encouraging result, considering the fact that there are no optimisation efforts made on the system. Such efforts would likely focus on preventing repetition of previously rejected expressions and would ultimately reduce the number of evaluations required.

The very fact that the techniques presented here have already been touched upon, in one form or another, as well as their conspicuous absence from the bulk

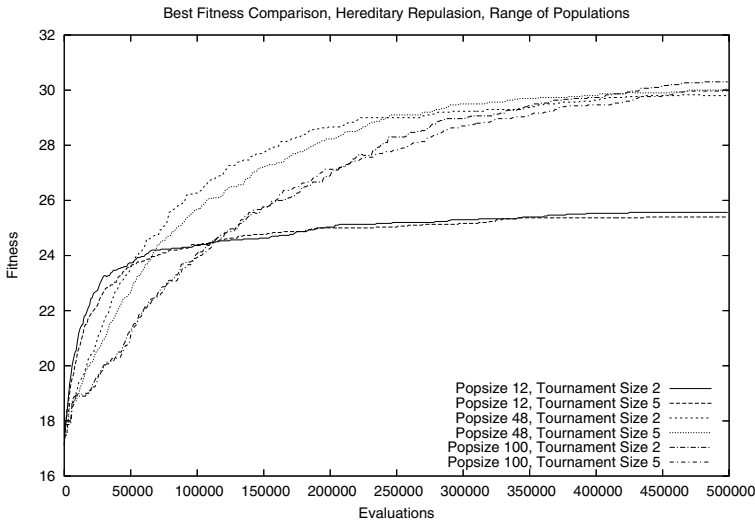


Figure 3-9. Best fitness averaged over 30 independent runs. Diagram clearly shows the significant improvement the hereditary repulsion model confers on the performance of the population. Note the high performance of the experiment using only a population size of 12. This clearly illustrates the capacity of HR to make the best use of the genetic content it has been given.

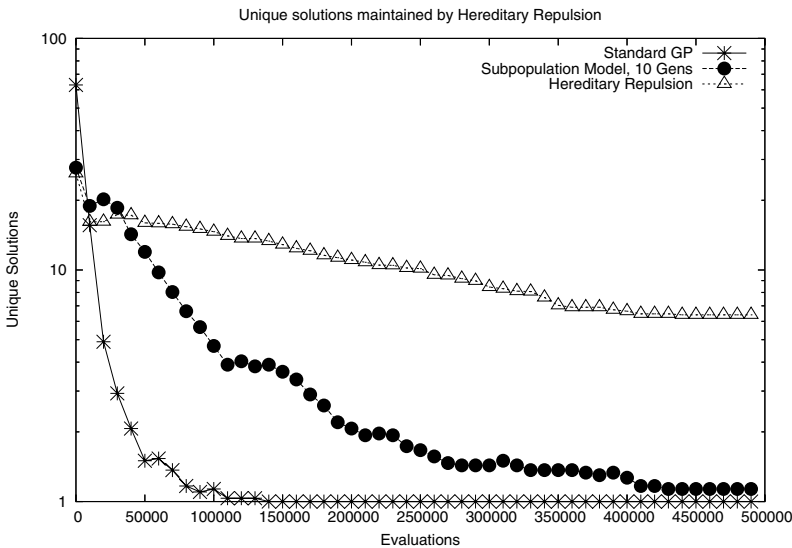


Figure 3-10. Solution densities for Hereditary Repulsion. The strength of the algorithm is seen in the way it successfully maintains a rich set of independent solutions over time, in contrast to the other methods

of the literature reflects the tendency of practitioners to answer the requirements of the problem of premature convergence with basic strategies described in Section 4.

The reason this is acceptable is that, by and large, such a strategy solves the problem at hand. The need for powerful algorithms such as these has become less pressing given the prevalence of powerful and cheap computing power. However, as the problems solved by the paradigm become more and more difficult, a well executed algorithm can preempt thousands of hours of computation.

The power behind evolutionary methods is well established within the Genetic Programming community and the onus is on us to proliferate the deployment of this technology throughout both industry and other scientific disciplines as a valid and powerful scientific methodology.

Analysis Techniques

Deducing the performance of the various techniques through the fitness graphs, while effective, is a very shallow evaluation of the algorithm. Future work will focus on developing efficient algorithmic techniques to map out the search space of the boolean domains. Whilst this represents a considerable technical challenge the benefits of such an analysis mechanism would be considerable.

Previous work done by the authors has found that artificially created problems created for analysing the behaviour of evolutionary models to be highly unstable. Problems were either too hard or too difficult. Because of this, the boolean domain presents a relatively difficult search space whose mapping is well within the capacity of modern computational hardware.

Having such an analysis mechanism would allow the observation of various critical sub solutions as they appeared with the expression space of the population. This would facilitate the further development of experimental hypotheses as the internal functioning of the model would be laid bare to the observer.

Further Development of Models

As previously mentioned, the models here were not fully developed to the extent of their potential. A viable source of future research would be the further development of these models.

The age based dynamics model has been very successful in (Hornby, 2005) and our experience with the model suggests that it certainly has a lot of potential for improvement.

The hereditary repulsion method has also been successfully tested on other unrelated domains, such as the BUPA liver disorders classification problem, as seen in Figure 3-11, however, an exhaustive decomposition of its core parameters has not yet been investigated. Of particular interest is the observation in

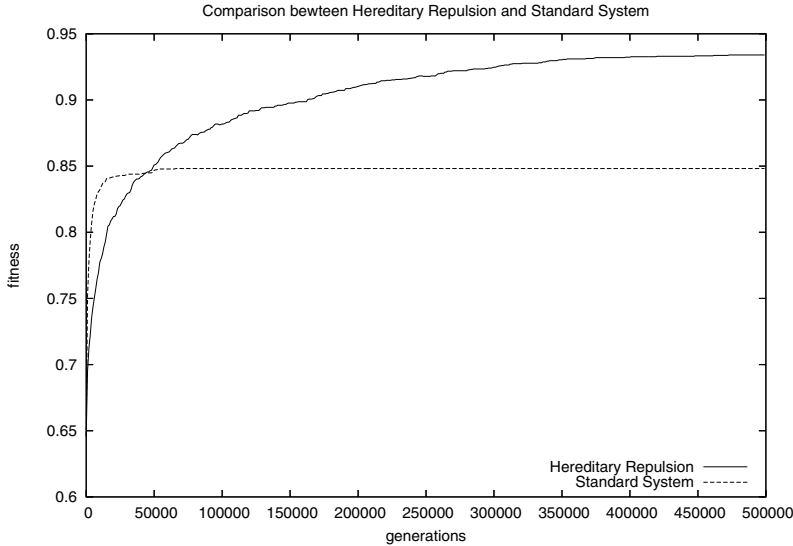


Figure 3-11. Best fitness averaged over 30 independent runs on the BUPA Liver Disorder Classification Problem. Figure compares performance of an Hereditary Repulsion model and standard model with the parameters described in Section 2. HR has performed well on a wide variety of problem domains.

individual runs of particularly high fitness individuals appearing in one cycle, only to disappear in the next.

Since the sampling rate is very high for each evolutionary cycle in Hereditary Repulsion, it is unlikely that the fit individuals have not been involved in crossover events. However, the constraint that the result of the crossover events must be better than *both* the parents seems the likely explanation. The disappearance of these high fitness individuals suggests that the system was not able to improve on them, and as such, represented an evolutionary deaden. An analysis technique such as the one previously described would verify the validity of this hypothesis.

9. Conclusions

This chapter has presented a number of techniques which have improved the performance of Genetic Programming based evolutionary models over standard evolutionary models. The primary purpose of this chapter was to detail the results of a series of experiments which employed variations of existing techniques within the science to improve the convergence dynamics of artificial evolutionary systems.

We have successful shown that by focusing on the requirement of balancing exploration with exploitation; the effectiveness of the algorithm can be signifi-

cantly improved. We hope that this has been of benefit to the community and will enhance the way they implement these algorithms.

References

- Braught, G.W. (2005). Learning and lineage selection in genetic algorithms. In *IEEE Transactions on Evolutionary Computation*. IEEE Computational Intelligence Society.
- D. Beasley, D. R. Bull and Martin, R. R. (1993). A sequential niche technique for multimodal function optimization. In *Evolutionary Computation*, volume 1, pages 101–125.
- Darwen, P.J. and Yao, Xin (1997). Speciation as automatic categorical modularization. In *IEEE Transactions on Evolutionary Computation*, volume 1. IEEE Computational Intelligence Society.
- Dehmeshki, J., Siddique, M., Lin, Xin-Yu, Roddie, M., and Costello, J. (2003). Automated detection of nodules in the ct lung images using multi-modal genetic algorithm. In *IEEE Transactions on Evolutionary Computation*, volume 1. IEEE Computational Intelligence Society.
- Goldberg, D. E. and Richardson, J (1987). Genetic algorithms with sharing for multimodal function optimisation. In *Proc. 2nd Int Conf. Genetic Algorithms*, pages 41–49.
- Holland, John .H. (1975). *Adaptation in Natural and Artificial Systems*. MIT Press.
- Hornby, Gregory S. (2005). Alps: the age-layered population structure for reducing the problem of premature convergence. In *Gecco 2005*.
- Hu, Jianjun and Goodman, Erik D. (2002). The hierarchical fair competition (HFC) model for parallel evolutionary algorithms. In Fogel, David B., El-Sharkawi, Mohamed A., Yao, Xin, Greenwood, Garry, Iba, Hitoshi, Marrow, Paul, and Shackleton, Mark, editors, *Proceedings of the 2002 Congress on Evolutionary Computation CEC2002*, pages 49–54. IEEE Press.
- Keijzer, Maarten, Ryan, Conor, Murphy, Gearoid, and Cattolico, Mike (2005). Undirected training of run transferable libraries. In Keijzer, Maarten, Tettamanzi, Andrea, Collet, Pierre, van Hemert, Jano I., and Tomassini, Marco, editors, *Proceedings of the 8th European Conference on Genetic Programming*, volume 3447 of *Lecture Notes in Computer Science*, pages 361–370, Lausanne, Switzerland. Springer.
- Koza, John R. (1994). *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge Massachusetts.
- Naoyuki Kubota, Toshio Fukuda, Fumihito Arai and Shimojima, Koji (1994). Genetic algorithm with age structure and its application to self organising manufacturing system. In *IEEE Symposium on Emerging Technologies and Factory Automation*.

Sareni, Bruno and Krahenbuhl, Laurent (1998). Fitness sharing and niching methods revisited. In *IEEE Transactions on Evolutionary Computation*, volume 2. IEEE Computational Intelligence Society.