Gradualism, neutrality and Binary Decision Diagrams: avoiding the exploration vs exploitation trade-off

xxxxxxxxxxxxxxxxxxxxxxxxxxxx

No Institute Given

Abstract. Investigation is presented into how neutrality can improve evolvability within the context of a new algorithm for evolving Binary Decision Diagrams. The perspective taken for the study is the familiar concept of the exploitation vs exploration relationship. The common perception of this relationship is that exploration must be balanced, or traded off, against exploitation using high mutation rates or population diversification methods. A key objective of this work is to better understand and cultivate this important relationship with regard to neutrality. The problem is first approached using a simple model of the search space, which is employed to generate reasoned arguments about its desirable properties. A metric, *adequacy*, is introduced for the relative potential of gradualism in a search space with respect to modality and mutation rate. The resulting hypothesis is that increased adequacy results in increased evolvability. This is tested empirically, and born out, using an experimental setup that effectively controls the source of exploration. The principle result is that, utilising neutrality, both exploration and exploitation can be encouraged *simultaneously*, avoiding the commonly perceived necessity for a trade-off. A further consequence is that parameters can be chosen with great confidence.

1 Introduction

Evolvability, the capacity to evolve, is an elusive but much sort after property. To this end, Kimura's neutral theory [5] has had a significant impact in evolutionary biology [11], and continues to be of interest to the EC community also [1, 4, 12]. However, there is debate as to whether neutrality can improve evolvability, and uncertainty as to how it does so [6, 10]. Therefore, understanding how neutrality can influence evolvability has important significance. This work takes a further step in this direction, building on Downing's[3] previous work with a new algorithm for evolving Binary Decision Diagrams (BDDs)[2] using implicit neutrality. The aim is to provide a better understanding of this algorithm's comparatively superior performance to other approaches, particularly with regard to neutrality and the minimal mutation rate employed.

The paper is organised as follows. Firstly, the concepts of exploration vs exploitation and gradualism are briefly reviewed. A description of the algorithm for evolving BDDs follows. A simple model of the search space is then presented, and the notion of *adequacy* introduced as a metric for the potential for gradualism within a search space. The hypothesis that greater adequacy leads to improved evolvability is then argued and tested experimentally. Discussion and summary conclude the paper. The reader should consult the stated work for complete details of the algorithm[3] and BDDs[2].

1.1 Exploration vs exploitation

Search algorithms, in general, suffer from problems associated with local optima. These problems severely limit the applicability of these algorithms, and attempts to address the problem are sometimes framed in the context of needing to balance exploration against exploitation. Ideally, it is best to maximise both simultaneously, but this is usually seen as infeasible in the presence of multi-modal landscapes. The assumed necessity for this trade-off is highlighted by Michalwicz & Fogel [8, p.45]:

> "How can we design a search algorithm that has a chance to escape local optima, to balance exploration and exploitation, and to make the search independent of the initial configuration?"

In the context of EC, exploitation is achieved by selecting the fitter parents from the population to breed (perhaps encouraged by elitist selection methods and smaller populations) and by using lower mutation rates to ensure that offspring inherit more of the characteristics of their parents. Conversely, exploration is achieved by promoting greater population diversity, and by increasing mutation rate. Promoting population diversity typically means having a larger population size and selecting parents less discerningly, perhaps employing mechanisms such as fitness sharing to encourage diversity. Increasing mutation rate means increasing the algorithm's proximity to random search, neglecting heredity. Clearly, attempting to achieve both exploitation and exploration simultaneously using these methods is antagonistic, hence the need for a trade-off or balance.

1.2 Gradualism

According to Mayr [7], Darwin's theory of evolution consists of five component theories: the non-constancy of species, common descent, natural selection, population thinking, and *gradualism*. Gradualism postulates that phenotypic differences between species arise gradually, in small steps, through a large number of intermediate forms. Darwin knew nothing of genetics, and formulated gradualism in respect of phenotypic change. However, gradualism can be recognised at the genetic level also where mutation rates are known to be very low, excessive mutations resulting in the phenomenon of *error catastrophe* or *extinction mutagenesis*. It stands to reason that gradual phenotypic change is facilitated by gradual genetic change, higher mutation rates producing greater proximity to random search.

2 Algorithm

Definition 1 *An* atomic mutation *changes the structure of the genotype in a manner that cannot be broken down into smaller steps which leave the OBDD still valid.*

The following atomic mutations are defined, four neutral and one adaptive, and applied to a population of OBDDs.[1] The adaptive mutation has the potential to change the function of the OBDD, the neutral mutations do not.

Definition 2 *Let* **N1** *be the neutral mutation of removing a redundant test. See figure 1(a).*

Definition 3 *Let* **N1'** *be the neutral mutation of inserting a redundant test, the inverse of* **N1***. See figure 1(a).*

Definition 4 *Let* **N2** *be the neutral mutation of removing a redundant non-terminal (merging two equivalent non-terminals). See figure 1(b).*

Definition 5 *Let* **N2'** *be the neutral mutation of inserting a redundant non-terminal, the inverse of* **N2** *(splitting). See figure 1(b).*

Definition 6 *Let* **A1** *be the adaptive mutation of changing one of the children of a non-terminal, to another vertex. See figure 2.*

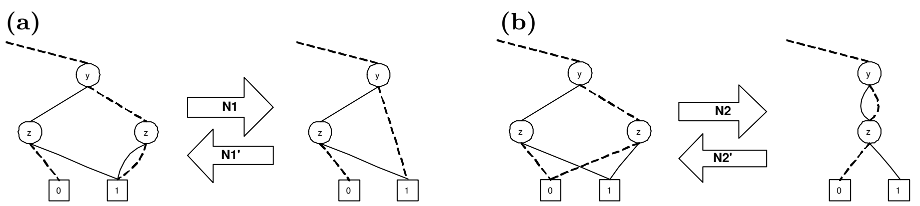

Fig. 1. Neutral mutaitons. (a) Remove/insert a redundant test. (b) Merge/split a non-terminal.

These mutations are applied to randomly selected vertices in a way that respects the requirement of total variable ordering demanded for all paths through an OBDD. The algorithm for evolving BDDs using implicit neutrality is described in figure 3.

[1] Please refer to Bryant [2] for definitions of BDD, *ordered* BDD (OBDD) and *reduced* OBDD (ROBDD).

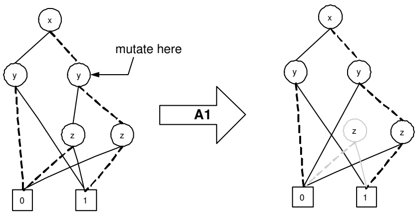

Fig. 2. A1 mutation

2.1 Local optima-free search for BDDs

This theorem, and its proof, was presented in Downing [3]. The theorem is repeated here for completeness.

Theorem 1. *Where all fitness increments are determined by individual input sets independently, and $fitness(f_{origin}) \leq fitness(f_{target})$, it is possible to transform any OBDD, $f_{origin}(x_0, ..., x_n)$, to any other OBDD, $f_{target}(x_0, ..., x_n)$, without loss of fitness in any of the successive intermediate OBDDs, using a series of atomic mutations only.*

3 Neighbourhoods, mutation relations and adequacy

A simple model of the search space is presented that existentially encapsulates possible transitions with respect to mutation rate, abstracting from the probabilities associated with particular transitions. Adequacy is then defined as a relative metric for the potential of gradualism within a search space with respect to modality and mutation rate. Finally, the adequacy of the search space associated with the algorithm presented in section 2 is determined, as is the necessity of the neutral mutations.

Definition 7 *The mutation rate, r, is an integer in the range* $[1, \infty[$*, capping the maximum number of atomic changes to the genotype from parent to offspring.*

Definition 8 *The* neighbourhood *of $g \in \mathcal{G}, \mathcal{N}_M^r(g) \subseteq \mathcal{G}$, consists of all those genotypes reachable from g, with mutation rate r and atomic mutation set M. $\mathcal{N}_M^1(g)$ is the* immediate neighbourhood *of g; $\mathcal{G}$ is the set of all genotypes.*

$$\mathcal{N}_M^r(g) = \bigcup_{i=1}^{r-1} \bigcup_{h \in \mathcal{N}_M^i(g)} \mathcal{N}_M^1(h)$$

Thus, $|\mathcal{N}_M^r(g)|$, for small r, can be much larger than $|\mathcal{N}_M^{r-1}(g)|$ because each member of $\mathcal{N}_M^{r-1}(g)$ potentially has an immediate neighbourhood size similar to the set from which it came. Furthermore, for small r, $\mathcal{N}_M^r(g)$ will typically be

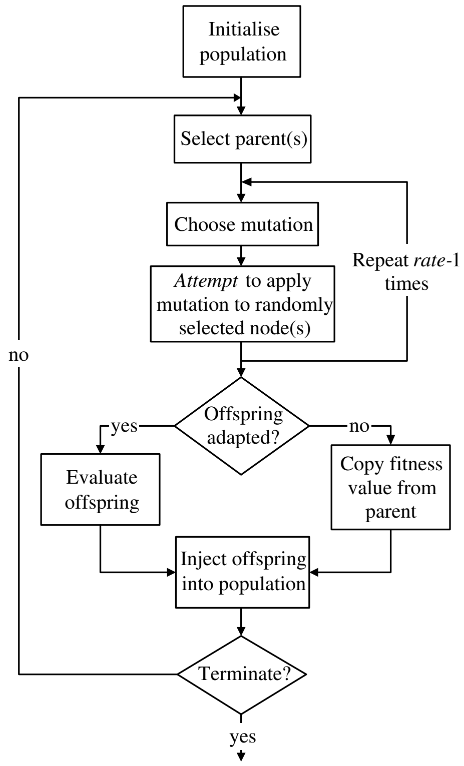

Fig. 3. Algorithm flowchart. It will often be the case that the selected vertex (or vertices) will not be amenable to the chosen mutation operation. For example, splitting (**N2'**) requires that a vertex have more than one incoming edge. In such cases, the mutation attempt is deemed a failure. If all *rate* mutation attempts are either failures or are neutral, the fitness value can be copied over from the parent because a neutral mutation does not affect function, giving an *effortless* neutral step on a neutral walk. This is in sharp contrast to other representations that employ neutrality. CGP [9], for example, typically requires a function evaluation after each mutation because it is not possible to determine in advance if a mutation will be neutral or adaptive.
Optionally, and in the following experiments, the population is periodically reduced every 8000 offspring. This aids parsimony of the population and tends to reduce overall computation time.

dominated by genotypes resulting from the more severe mutations. However, as r increases, the potential for overlap increases leading to fewer and fewer new genotypes being incorporated into the neighbourhood.

Definition 9 *The* extended neighbourhood *of $g \in \mathcal{G}$, denoted $\mathcal{N}\S^r_M(g)$, incorporates all the neighbourhoods of g's neutral network.*

$$\mathcal{N}\S^r_M(g) = \bigcup_{h \in \mathcal{G} | h \equiv g} \mathcal{N}^r_M(h)$$

In effect, $\mathcal{N}\S^r_M(g)$, defines the neighbourhood of a neutral network. $|\mathcal{N}\S^r_M(g)|$ will typically be larger where g has more functional equivalents than fewer. Furthermore, functions with more compact ROBDDs typically have more functionally equivalent OBDDs than less compact ROBDDs. Thus, $|\mathcal{N}\S^r_M(g)|$ is typically larger where g represents a function with a compact ROBDD.

Definition 10 *The* mutation relation*, $\mathcal{M}_{M^r} \subseteq \mathcal{G} \times \mathcal{G}$, relates genotypes to genotypes in respect of all neighbourhoods.*

$$\mathcal{M}_{M^r} = \bigcup_{g \in \mathcal{G}} \bigcup_{h \in \mathcal{N}^r_M(g)} \{(g, h)\}$$

Definition 11 *$\mathcal{M}$ is said to be* adequate*, denoted $\mathcal{M}^A$, if it provides a search space free of local optima for a given problem.*[2] *That is, for each $g_{origin} \in \mathcal{G}$, there is a series of elements:*

$$(g_{origin}, g_i), (g_i, g_j), (g_j, g_k), \ldots, (g_{target-2}, g_{target-1}), (g_{target-1}, g_{target})$$

where g_{target} represents the global optimum.

Definition 12 *$\mathcal{M}$ is said to be* completely adequate*, or simply* complete*, denoted $\mathcal{M}^C$, if $\mathcal{M} = \mathcal{M}^A_{M^1}$. $\mathcal{M}$ is* universally complete*, denoted $\mathcal{M}^U$, if it is complete for all problems within a specified domain.*

$\mathcal{M}^A$ is easy to achieve, simply by allowing mutation rates high enough to approximate random search. However, high mutation rates defy gradualism and are not conducive to evolvability. A more useful property is *complete adequacy*, or *completeness*, $\mathcal{M}^C$. While $\mathcal{M}^C$ suggests maximum adherence to the concept of gradualism, $\mathcal{M}^A$ alone does not. This suggests an ordering of increasing adequacy based on decreasing mutation rate whilst maintaining a local optima free search space.

Definition 13 *The* adequacy ordering*,*

$$\mathcal{M}^C_M, \mathcal{M}^A_{M^2}, \ldots, \mathcal{M}^A_{M^\infty}$$

is an ordering of mutation relations of decreasing adequacy.

Adequacy, whether complete or not, does not necessarily imply optimality, hence its name. Adequacy simply gives a way to reason about the relative degree of gradualism facilitated within $\mathcal{M}$, whilst maintaining a level playing field in respect of local optima and the atomic mutation set.

[2] The subscript is omitted for brevity.

3.1 Adequacy for evolving BDDs

Corollary 1. $\mathcal{M}_{\{N1',N2',A1\}^1}$ *is universally complete.*

This follows from the fact that these are the only atomic mutations required by theorem 1 to prove that the search space is free of local optima for the specified problem domain [3].

Corollary 2. $\{$***N1'***, ***N2'***, ***A1***$\}$ *is the minimal atomic set that will produce* $\mathcal{M}^U$.

Removing **A1** means that an initial population that does not contain an optimal genotype cannot be functionally transformed to one that does. Removing **N1'** or **N2'** means that an initial population of individuals that do not have sufficient nodes at a given level cannot insert any new nodes. Thus, neutral mutations are essential to this algorithm.

Corollary 3. $\mathcal{M}_{\{N1,N1',N2,N2',A1\}^1}$ *is universally complete in the context.*

Adding atomic mutations to the minimal set does nothing to improve or detriment the adequacy of $\mathcal{M}$, as the adequacy ordering relates only mutation relations with identical M. However, adding atomic mutations may impact on the performance of the algorithm. In this case, the addition of the reducing neutral mutations **N1** and **N2**, encourage parsimony and modularity in the genotype. Performance can also be influenced, without affecting adequacy, by biasing mutation selection, potentially influencing the ratio of adaptive to neutral mutations.

3.2 Model discussion

As a run progresses towards optimum fitness, the neighbourhood and extended neighbourhood become increasingly dominated by less fit solutions: This is largely why EAs exhibit a performance graph of fitness against effort which levels off rapidly as the run progresses towards the optimal, and improved solutions become increasingly harder to find. For mutation rates greater than 1, each intermediate atomic mutation has a much higher probability of being deleterious than advantageous. Furthermore, the degree to which an intermediate atomic mutation can be deleterious outweighs the degree to which it can be advantageous. For higher mutation rates, therefore, a successful early intermediate atomic mutation is increasingly likely to be undone by a later one. Similarly, a successful later intermediate atomic mutation is likely to be preempted by an earlier deleterious one. As a result, for higher mutation rates, it is increasingly likely that an overall deleterious mutation will occur.

Important in the model is the definition of mutation rate, where an integer caps the maximum number of atomic mutations. For example, in a bit-string representation, this integer would refer to the maximum number of bit flips. This definition of mutation rate is distinct from the one that would mutate each bit with a certain probability. This latter definition relates mutation rate to the maximum potential degree of genotypic change probabilistically, not existentially: Thus, altering the mutation rate would have no impact on the mutation relation.

Mutation rate, with respect to the algorithm presented in section 2, actually refers to mutation attempts, each of which will often fail, i.e. a mutation rate of 3 means 3 attempts to mutate, with a probability of 3 actual atomic mutations typically much less than certain. For the algorithm presented in section 2 this probability is dependant on the given genotype, but approximates to about one third per mutation attempt.

3.3 Hypothesis

The hypothesis predicts, for contrast and comparison, performance characteristics of two versions of the algorithm; the one presented in section 2, and a modified form that simulates the need to evaluate *both* neutral and functionally distinct offspring (but not clones), as is necessary for the majority of EAs. This second algorithm is referred to as the *impaired* form of the algorithm, and its inclusion aids understanding of the relationship between the ratio of adaptive to neutral mutations, mutation rate, and evolvability in the general case. Similarly, the former is referred to as the *unimpaired* form.

The prediction is that increased adequacy results in increased evolvability: maximal adequacy, or completeness, provides maximum evolvability. More formally:

$$AES(\mathcal{M}_M^C) \gtrapprox AES(\mathcal{M}_{M^2}^A) \gtrapprox \ldots \gtrapprox AES(\mathcal{M}_{M^\infty}^A)$$

where AES[3] is the evolvability performance measure. For the unimpaired algorithm, the prediction is expected to hold for any bias that is introduced for mutation selection, affecting the ratio of adaptive to neutral mutations but not the adequacy. Any increase in neutral drift due to the lowering of this ratio only provides an increase in exploration, and it does so without cost.

However, for the impaired form of the algorithm, the prediction is particularly tentative because it assumes an optimal ratio of adaptive to neutral mutations, for which an optimal bias in mutation selection must be achieved. Too low a ratio and exploitation is neglected in favour of evaluating an excessive number of neutral offspring. Conversely, too high a ratio, and exploration is neglected, excessive time being spent evaluating solutions already seen. Both situations are expected to result in inflated AES, particularly around the lower mutation rates for the former case, forming a trough in the performance graph of AES against mutation rate, breaking the above prediction. However, increasing the ratio through biasing in favour of adaptive mutation is expected to shift the trough toward the vertical axes, eventually eliminating it, improving on the overall optimum which now should reside at the minimal mutation rate.

[3] Average number of Evaluations to a Solution.

4 Experiments

For the following experiments, a (1 + 1) ES eliminates any potential for exploration through population diversity, but facilitates maximum exploitation in terms of always breeding the fittest individual seen so far. Biasing mutation selection influences the ratio of adaptive to neutral mutation, without consequence to adequacy.

4.1 Unimpaired algorithm

The hypothesis is tested here on the algorithm of section 2. The results for several parity and multiplexer problems are shown in figure 4.

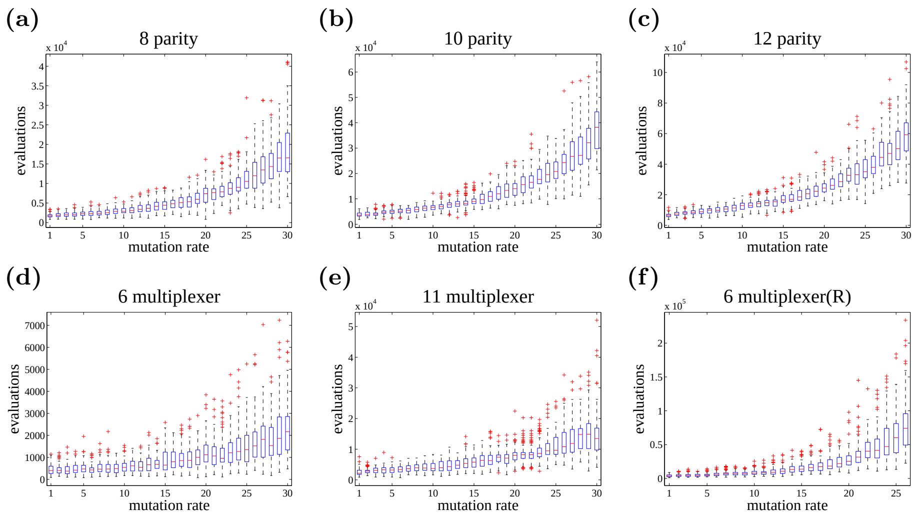

Fig. 4. Box and whisker plots for the unimpaired algorithm (100 runs). (a)-(c) parity problems. (d) and (e) multiplexer problems with optimal variable orderings. (f) is 6 multiplexer with reverse variable ordering. For all problems, optimal performance is achieved at minimal mutation rate. AES and variance increases rapidly with mutation rate. The results are consistent with the hypothesis.

As can be seen, increasing mutation rate typically results in a decrease in performance in terms of both increased AES and variance. These results are consistent with the hypothesis' prediction. Figure 5 examines the effects of introducing a bias to mutation selection. The behaviour is largely as predicted. Increasing the ratio of adaptive to neutral mutations, stifling neutral drift, typically results in an increase in AES. Conversely, lowering the ratio is synonymous with increasing neutral drift without cost, and thus exploration, resulting in lower AES.

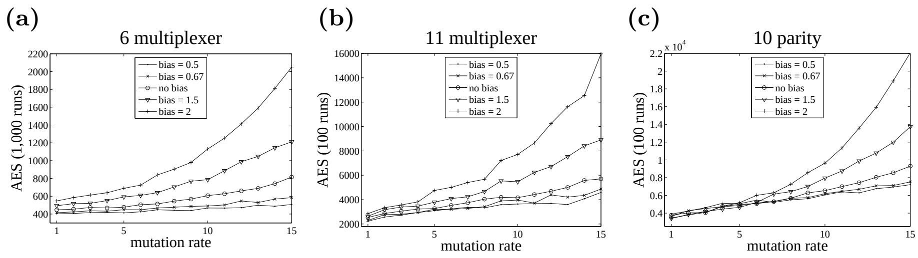

Fig. 5. The effects of introducing mutation selection bias for selected problems. The probability of selecting an adaptive mutation is multiplied by the bias. Increasing mutation rate typically results in increased AES, regardless of bias, as predicted. In (a) and (b), decreasing the adaptive to neutral ratio through a <1 bias results in an overall increase in performance for all mutation rates; increasing the ratio decreases performance. (c) is a less conclusive, but the deviation from the behaviour expected is minimal.

4.2 Impaired algorithm

The impaired algorithm is investigated here for contrast and comparison with the unimpaired form, and to simulate the need to evaluate both adaptive and neutral offspring as is necessary for the majority of EAs. Figure 6 shows three very different behaviours for biasing the adaptive to neutral ratio of mutations on the impaired algorithm. The three behaviours also differ from that of the unimpaired algorithm.

Firstly, excessive neutral drift can be observed where a trough is formed. In both figures 6(a) and 6(b) the trough is formed without bias, but 6(c) demonstrates how such a trough can be introduced by decreasing the ratio of adaptive to neutral mutations, causing excessive neutral drift. This behaviour deviates from that of the unimpaired algorithm, where increasing neutral drift is not detrimental because it is achieved without cost.

Secondly, attempting to achieve an optimal ratio of adaptive to neutral mutations is problematic. Where a trough exists, reducing neutral drift with an increased bias is expected to shift the trough toward the vertical axes, improving on the optimal as it goes. This is indeed the behaviour exhibited in both 6(b) and 6(c). However, in contrast, this it is not the behaviour exhibited in 6(a). While the trough does indeed move toward the axes as the ratio is increased, performance deteriorates universally across mutation rates.

Finally, where no trough exists, increasing the ratio is expected to deteriorate performance universally, as for the unimpaired algorithm, because exploration is stifled. This is indeed the behaviour exhibited in 6(b), but is not the behaviour exhibited in 6(c), which actually sees an improvement in performance at lower mutation rates.

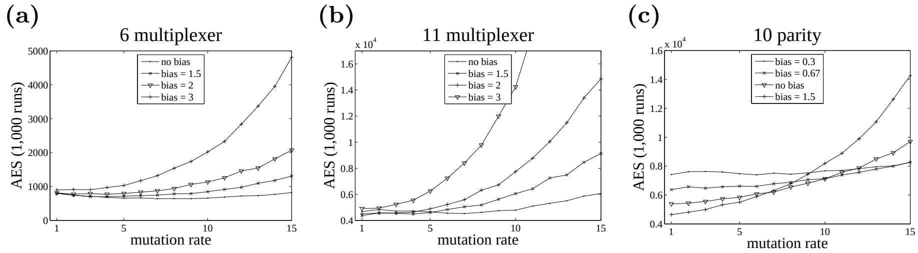

Fig. 6. Three differing effects of introducing a selection bias for the impaired algorithm. (a) No bias results in the optimal residing away from the minimal mutation rate, forming a trough. Increasing the bias shifts the optimal towards the vertical axes, eliminating the trough, but fails to improve overall AES. (b) For no bias, a trough is formed. Increasing the bias shifts the optimal towards the axes, but this time improves overall optimal AES up to a point. (c) For no bias, no trough is formed, indicating no excessive neutral drift. Introducing more neutral drift increases AES, introducing a trough. Reducing neutral drift unexpectedly improves overall optimal AES.

5 Conclusion

These results challenge the commonly perceived necessity for a trade-off, or balance, between exploration and exploitation [8, p.45]. The unimpaired algorithm requires no trade-off because exploration can be increased, without cost, by introducing a <1 mutation selection bias. The impaired algorithm *does* require a trade-off in terms of the ratio of adaptive to neutral mutations and mutation rate, but avoids the need to employ high, antagonistic mutation rates or population diversity techniques purported to be able to offer worthy alternatives to neutrality for exploration[6].

In practice, this understanding facilitates some certainty in the choice of parameters. Where the fitness function is in the domain of theorem 1, as will most typical classification type fitness functions will, minimal mutation rate and population should be employed. The bias of the ratio of adaptive to neutral mutations is the only other parameter to consider, its effects being fairly predictable. Even with the impaired algorithm, there can be a high degree of confidence in a lower mutation rate.

6 Summary

An investigation has been presented into how neutrality facilitates evolvability within the context of a new algorithm for evolving BDDs. The algorithm facilitates a local optima-free search space for a limited, but significant, domain of fitness functions. Neutral mutations are a necessity for this property. A simple model of the search space facilitates reasoning about the degree of gradualism facilitated. The conclusion is that neutrality can effectively facilitate both exploration and exploitation simultaneously, avoiding the antagonisms associated high mutation rates and large populations, and facilitating great confidence in the choice of parameters. Future work will focus on further extending and applying Downing's algorithm, and further understanding neutrality within its context.

References

1. Lionel Barnett. Netcrawling - optimal evolutionary search with neutral networks. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, pages 30–37, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, 27-30 May 2001. IEEE Press.
2. Randal E. Bryant. Graph-Based Algorithms for Boolean Function Manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, August 1986.
3. Richard Mark Downing. Evolving Binary Decision Diagrams using Implicit Neutrality. In *The 2005 IEEE Congress on Evolutionary Computation, CEC 2005*, pages 2107–2113, Edinburgh, Scotland, UK, September 2005. IEEE Press.
4. Marc Ebner, Patrick Langguth, Juergen Albert, Mark Shackleton, and Rob Shipman. On neutral networks and evolvability. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, pages 1–8, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, 27-30 May 2001. IEEE Press.
5. Motoo Kimura. *The neutral theory of molecular evolution.* Cambridge University Press, 1983.
6. Joshua D. Knowles and Richard A. Watson. On the utility of redundant encodings in mutation-based evolutionary search. In H.-G. Beyer J.-L. Fernández-Villacañas H.-P. Schwefel J.-J. Merelo Guervós, P. Adamidis, editor, *Parallel Problem Solving from Nature - PPSN VII, 7th International Conference, Granada, Spain, September 7-11, 2002. Proceedings*, number 2439 in Lecture Notes in Computer Science, LNCS, page 88 ff. Springer-Verlag, 2002. Keywords: Technique::Comparisons of representations, Theory of EC::Fitness landscape.
7. Ernst Mayr. *What Evolution is.* Phoenix, 2002.
8. Zbigniew Michalewicz and David B. Fogel. *How to Solve It: Modern Heuristics.* Springer-Verlag, New York, 2000.
9. Julian F Miller and Peter Thomson. Cartesian genetic programming. In R. Poli, W. Banzhaf, W. B. Langdon, J. Miller, P. Nordin, and T. C. Fogarty, editors, *Proceedings of the Third European Conference on Genetic Programming (EuroGP-2000)*, volume 1802 of *LNCS*, pages 121–132, Edinburgh, Scotland, 2000. Springer Verlag.
10. Franz Rothlauf and David E. Goldberg. Redundant representations in evolutionary computation. *Evol. Comput.*, 11(4):381–415, 2003.
11. Andreas Wagner. *Robustness and Evolvability in Living Systems.* Princeton Studies in Complexity. Princeton University Press, 2005.
12. Tina Yu and Julian Miller. Finding needles in haystacks is not hard with neutrality. In J. A. Foster, E. Lutton, J. Miller, C. Ryan, and A. G. B. Tettamanzi, editors, *Proceedings of the Fifth European Conference on Genetic Programming (EuroGP-2002)*, volume 2278 of *LNCS*, pages 13–25, Kinsale, Ireland, 2002. Springer Verlag.