Scalability of Selectorecombinative Genetic Algorithms for Problems with Tight Linkage

Kumara Sastry[1,2] and David E. Goldberg[1,3]

[1] Illinois Genetic Algorithms Laboratory (IlliGAL)
[2] Department of Material Science & Engineering
[3] Department of General Engineering
University of Illinois at Urbana-Champaign, Urbana, IL
{ksastry,deg}@uiuc.edu

Abstract. Ensuring building-block (BB) mixing is critical to the success of genetic and evolutionary algorithms. This study develops facetwise models to predict the BB mixing time and the population sizing dictated by BB mixing for single-point crossover. The population-sizing model suggests that for moderate-to-large problems, BB mixing – instead of BB decision making and BB supply – bounds the population size required to obtain a solution of constant quality. Furthermore, the population sizing for single-point crossover scales as $O\left(2^k m^{1.5}\right)$, where k is the BB size, and m is the number of BBs.

1 Introduction

Since the inception of genetic algorithms (GAs), the importance of building blocks (BBs) has been recognized [1,2]. Based on Holland's notion of BBs, Goldberg proposed a design decomposition method for a successful design of GAs [3]. This design decomposition currently consists of sevens steps [4] and can be stated as follows: (1) Know what GAs process – building blocks (BBs), (2) solve problems that are of bounded BB difficulty, (3) ensure an adequate supply of raw BBs, (4) ensure increased market share for superior BBs, (5) know BB takeover and convergence times, (6) ensure that BB decisions are well made, and (7) ensure a good mixing of BBs. Significant progress has been made in developing facetwise models for many of the above decomposition steps and the interested reader should consult [4] for further details.

However, researchers have often overlooked the issues of BB identification and mixing or exchange, even though studies on selectorecombinative GAs have indicated that effective *identification* and *exchange* of BBs is critical to innovative success. Furthermore, existing facetwise models such as convergence-time and population-sizing models *assume* tight linkage. That is, alleles of a BB are assumed to be close to one another, and crossover operators are assumed to ensure necessary exchange of BBs with a high probability. Even though the tight-linkage assumption isolates the phenomenon of interest while bracketing the linkage problem, in real-world problems this is not the case, as we don't know which alleles contribute to which BBs.

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2724, pp. 1332–1344, 2003.

It is therefore critical to understand mixing capability of popular recombination operators used in genetic and evolutionary algorithms. Dimensional models for BB mixing have been developed for uniform crossover [5,6] and similar analysis is yet to be done on single-point and two-point crossovers [7]. For problems with loosely linked BBs, mixing behavior of multi-point crossover is bounded by the mixing model of uniform crossover. On the other hand, for problems with tightly linked BBs, mixing behavior of multi-point crossovers can be different from that of uniform crossover and separate dimensional models have to be developed. Therefore the objective of this study is to develop a facetwise model to predict the mixing behavior of single-point crossover and to utilize this model to predict the mixing time and population sizing dictated by BB mixing.

This paper is organized as follows: The next section presents a brief literature review. Section 3 defines the mixing problem and states the assumptions used in developing the facetwise models. Section 4 develops mixing models for problems with two building blocks, which is extended for a more general m-BB case in Sect. 5. Finally, the paper is summarized and key conclusions are stated.

2 Literature Review

Many researchers have analyzed different aspects of fixed point crossovers and the efforts can be broadly classified into four categories [7]:

Recombination Modeled as Schema Disrupter: This class of models is motivated from the schema theorem [1]. These studies model schemata disruption of different crossover operators and are useful in comparing those operators [8,9,10]. However, as shown elsewhere, these models do not address the mixing problem [11].

Recombination Modeled as Mixer: Models belonging to this class are motivated from quantitative genetics and quantify properties such as *linkage disequilibrium* [12,13], or crossover-induced bias such as *positional bias*, *distributional bias*, *recombinative bias* and *schema bias* [14,13,15], and *relaxation time* [16]. However, these models also do not address the mixing issue.

All-in-One Models: This class of models combines all the facets of recombination operators using tools such as difference equations, Markov chains, and statistical mechanics [17,18,19,20]. Such models are more accurate and contain all the information about BB mixing, however, it is very hard to extract the BB-mixing information out of such complex models.

Recombination Modeled as an Innovator: This class of models [5,6,21] address the BB mixing issue in a more direct manner than the above mentioned class of models . These models predict *mixing* or *innovation* time, which is defined as the expected number of generations to obtain an instance of the target solution. These models have been compared with other facetwise models using dimensional arguments and *control maps* have been constructed. Such a control map identifies different competing forces affecting the genetic search. However, these models have been developed only for uniform crossover. While the mixing model for uniform crossover bounds the behavior of multi-point crossover for problems with loosely linked BBs, their mixing behavior can be significantly different from that of uniform crossover for problems with tightly linked BBs. Therefore, we analyze the mixing behavior of one-point crossover for problems with tight linkage.

3 Problem Definition

The section describes the mixing problem and the assumptions made to facilitate its analysis. Before doing so, it should be noted that there are two ways a crossover can increase the number of BBs on a particular string. One possibility is that the BB at a certain position is *created*. However, the likelihood of BBs being created decreases as the BB size increases. The other possibility is that the BBs are *mixed*. That is, the crossover operator transfers BBs at different positions from both parents to one of the offspring. For instance, a crossover can combine the following two strings, `bb####` and `##b###`, to yield following two offspring, `bbb##`, and `######`. Here `b` refers to a BB and `#` refers to schemata other than the BB.

This aspect of crossover – its ability to recombine BBs in order to better solutions – is of particular interest to us. Specifically, the rate at which a recombination operator exchanges BBs dictates the success of a GA run. Therefore, we model the mixing rate of single-point crossover on search problems with tightly linked BBs. Specifically, we answer the following question: Given that the individuals in the population have m_c BBs, how long – in terms of number of generations – will it take to obtain individuals with $m_c + 1$ BBs. This time is defined as the *mixing time* and is denoted by t_x.

To ease the analytical burden, we consider generation-wise selectorecombinative GAs with non-overlapping population of fixed size. The decision variables are assumed to be encoded into a binary string of fixed length. Furthermore, we consider the class of search problems with uniformly-scaled BBs. Uniform scaling implies that the contribution of BBs from different partitions to the overall fitness of the individual is the same. Specifically, we consider fully-deceptive trap functions [22] to validate the mixing models. However, the results should apply to additively decomposable stationary fitness functions of bounded order [4].

Using these assumptions, the following section develops a dimensional model for mixing time. The mixing-time model, and a convergence-time model will be used to develop a population-sizing model dictated by BB mixing. Our approach is methodologically similar to that of Thierens and Goldberg [6], in that we consider the mixing of a simple two building block problem and later extending it to a more general m building block case. Interestingly, the similar approach leads to radically different models as should be expected. In the original study, exponential times were required to solve uniformly mixed deceptive problems. Here, polynomial time models are predicted and observed for tight linkage with single-point crossover. The ability to derive such different models accurately is testimony to the analytical power of the approach.

4 Building-Block Mixing Models: Two BBs

This section presents dimensional models to predict the mixing behavior of single-point crossover on a problem with two building blocks, each of size k. Here, we assume that individuals in the initial population have exactly one BB. Therefore, in the two-BB case, a single mixing event mixing two BBs from different partitions to yield the global optimum. That is, if the following two individuals are selected for crossover: `b#` and `#b`, and the crossover site is chosen as k, then one of the offspring is `bb`, which is the global optimum. Recognizing that a mixing event can occur in two scenarios (`b#`+`#b`, or `#b`+`b#`) out of the possible four crossover scenarios and if the crossover site is chosen between the BBs (crossover site is k), we can write probability of a mixing event as

$$p_{\text{mix}} = \left(\frac{2}{4}\right)\left(\frac{1}{2k-1}\right). \tag{1}$$

Equation (1) assumes that every individual in the population contains either a BB (all ones) or a deceptive attractor (all zeros). However, in the actual case, the probability of having a BB in a string is 2^{-k}, and not having a BB is $2^{-k}\left(2^k-1\right)$. Therefore, the proportion of recombinations that can result in a mixing event is given by $2^{-2k}\left(2^k-1\right)$. In such a case probability of mixing is given by

$$p'_{\text{mix}} = 2 \cdot 2^{-2k}\left(2^k-1\right)\left(\frac{1}{2k-1}\right) \approx \frac{2^{-(k-1)}}{2k-1}. \tag{2}$$

Assuming a population size, n and a crossover probability, p_c, there are $\frac{n}{2}\cdot p_c$ recombinations in one generation. Therefore, the mixing time, t_x is

$$t_x = \frac{1}{\frac{n}{2}p_c p'_{\text{mix}}} = \frac{2^k(2k-1)}{np_c}. \tag{3}$$

The above mixing time model is compared with the following convergence-time model [23,24,25] to interrelate recombination time with selection time:

$$t_{\text{conv}} = \frac{c_c\sqrt{\ell}}{I} \tag{4}$$

where, I is the selection intensity, which is a function of tournament size, and c_c is a constant. Recognizing that for innovative success the mixing time has to be less than the convergence time (recombination has to create the global optimum before the population converges to the local optimum), we can write, $t_x < t_{\text{conv}}$.

For the two-BB case, $\ell = 2k$, and the above equation can be written as

$$n > \frac{I}{c_c p_c}\left(\frac{2^k(2k-1)}{\sqrt{2k}}\right) \approx c_{x2}\frac{I}{p_c}\left(2^k\sqrt{k}\right), \tag{5}$$

where $c_{x2} = 2\left(c_c\right)^{-1}$ is a constant. Equation (5) predicts the population size dictated by BB mixing. It is verified with empirical results in Fig. 1. The empirical

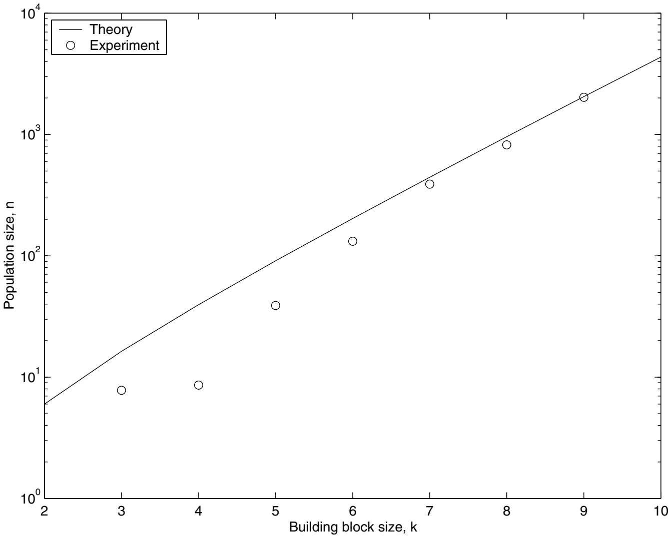

Fig. 1. Empirical validation of the population-sizing model dictated by BB mixing (Eq. (5)). Minimum population size required for a GA success is plotted as a function of BB size, k. The empirical population size is determined by a bisection method and are averaged over 25 independent bisection runs

results are obtained as follows: A binary tournament selection and single-point crossover with $p_c = 1$ is used. A GA run is terminated when all the individuals in the population converge to the same fitness value. The average number of BBs correctly converged are computed over 50 independent runs. The minimum population required such that both BBs converge to the correct value is determined by a bisection method. The results shown in Fig. 1 are averaged over 25 such bisection runs.

5 Building-Block Mixing Models: m BBs

To facilitate the analysis to m building blocks, we blend empirical observations with theory similar to the approach followed by Thierens and Goldberg [6]. The empirical results on m k-bit deceptive traps indicates two key behaviors of single-point crossover:

1. All the building block configurations at a certain *mixing level* have to be discovered before a higher mixing level can be reached. Here a mixing level denotes the number of BBs present in an individual string. For instance, individuals `b#b##`, and `b#bb#` are at mixing levels 2 and 3, respectively. This *ladder-climbing* phenomenon is illustrated in Fig. 2, which plots the proportion of all possible BB configurations at different mixing levels as a function of time. It can be clearly seen that when a certain mixing level is at

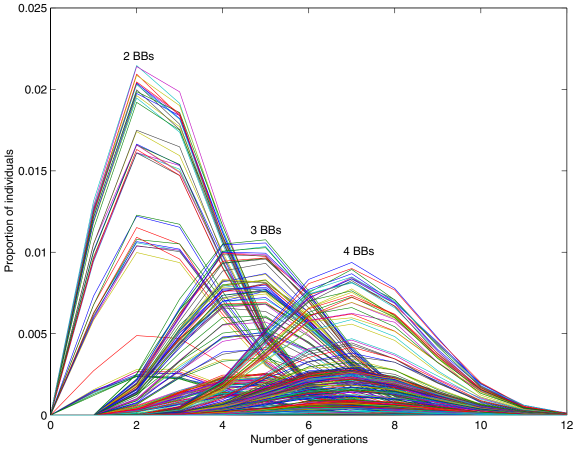

Fig. 2. Proportion of individuals with correct BBs as a function of time. Empirical results indicate that mixing at a higher level ($m_c + 1$) takes over only after BBs of lower level (m_c) are covered. The results are for ten 4-bit deceptive trap functions

its maximum value – when the proportion of all the BB configurations at a certain mixing level in the population is highest – proportion of individuals at higher mixing level is negligible, and hence, we can use the *ladder-climbing* model [6].

2. The proportion of individuals having good BBs at their two ends is higher than those at other positions. In fact, the proportion gradually reduces along the position and reaches a minimum at the center, as illustrated in Fig. 3. The figure plots the proportion of individuals having a good BB as a function of its position in the string and time. This is because the probability of mixing is much higher when mixing two BBs at the extreme points in the string. For example, the mixing probability when recombining `b#···##` and `##···#b` to yield `b#···#b` is $((m-2)k+1)/(mk-1)$ as opposed to recombining `b##···#` and `#b#···#` to yield `bb#···#`, in which case the mixing probability is $1/(mk-1)$.

   This property of single-point crossover is called the *length-reduction* phenomenon and is what makes the mixing behavior of one-point crossover different from that of uniform crossover. An immediate consequence of the length-reduction phenomenon is that an increase in mixing level leads to a decrease in problem size. For example, when we are at mixing level 3, the problem size reduces by two. This is because the BBs at the two ends get fixed for majority of the individuals.

The combined effect of *ladder-climbing* and *length reduction* illustrated in Fig. 4. The length-reduction phenomenon suggests that we can consider mixing probability as a function of number of BBs that are not converged at a given time and the BB size. Furthermore, we assume that among the BBs that are not converged the mixing level is always one.

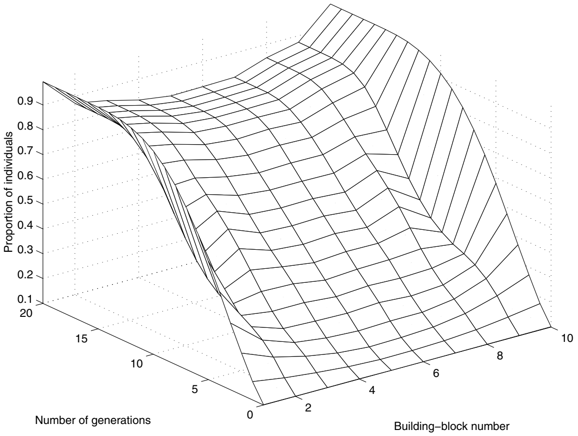

Fig. 3. Proportion of individuals with correct BBs as a function of BB position and time. Mixing of one-point crossover is dependent on the BB position. It is highest on the string ends and lowest in the middle of the string. The results shown are for a 10 4-bit deceptive trap functions

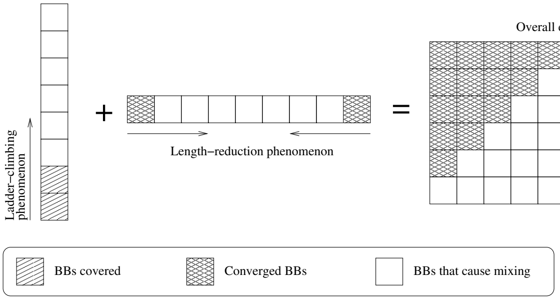

Fig. 4. Illustration of the *ladder-climbing* and *length-reduction* phenomena and their combined effect. Ladder-climbing phenomenon suggests that we increase the mixing level one step at a time. Length-reduction phenomenon suggests that as the mixing level increases, the problem size dictating mixing shrinks. It also suggests that the string ends converge faster than the BBs at the middle of the string

Given that a problem has m BBs and we are currently at mixing level one, that is, every individual has one BB at some position. Out of a total of m^2 recombination scenarios, the following result in mixing:

Recombination scenario	b##···## #b#···##	b###···# ##b#···#	···	b#··· ##···# ##···b#···#	···	b#··· ## ##···#b
Possible events	$2(m-1)$	$2(m-2)$	$\cdots$	$2(m-i)$	$\cdots$	2
Mixing Probability	$\frac{1}{mk-1}$	$\frac{k+1}{mk-1}$	$\cdots$	$\frac{(i-1)k+1}{mk-1}$	$\cdots$	$\frac{(m-2)k+1}{mk-1}$

Therefore the overall mixing probability can be written as

$$p_{\text{mix}}(m) = \frac{2}{m^2} \sum_{i=1}^{m-1} \frac{(m-i)\,((i-1)k+1)}{mk-1}, \tag{6}$$

which can be simplified as follows:

$$p_{\text{mix}}(m) = \frac{2}{3} \cdot \frac{(m-1)\,[(m-2)k+3]}{m(mk-1)} \tag{7}$$

From the length-reduction phenomenon, we know that, at a given mixing level, m_c, the mixing probability is given by

$$p_{\text{mix}}|_{m_c} = p_{\text{mix}}\left(m - m_c + 1\right) = \frac{2}{3}\left[\frac{(m-m_c)\,((m-m_c-1)k+3)}{(m-m_c+1)\,((m-m_c+1)k-1)}\right]. \tag{8}$$

For moderate-to-large problems, the mixing probability can be assumed to be constant with respect to the mixing level and can be approximated as follows:

$$p_{\text{mix}} \approx p_{\text{mix}}\left(m_c = 1\right) = \frac{2}{3}\left[\frac{(m-1)\,((m-2)k+3)}{m(mk-1)}\right]. \tag{9}$$

Now the question remains as to how we reach mixing level m_c+1 when we are at mixing level m_c. Assume that after n_x mixing events, we have n_x individuals at mixing level m_c+1. We have to find out how many mixing events are needed to have all BBs covered at mixing level m_c+1. Since all the n_x strings have m_c+1 BBs, it can be expected that $m\left[1-\left(1-\frac{m_c+1}{m}\right)^{n_x}\right]$ BBs are covered. The probability that all BBs are covered approaches 1, and the number of mixing events, n_x is increased.:

$$m\left[1-\left(1-\frac{m_c+1}{m}\right)^{n_x}\right] > m(1-\alpha), \tag{10}$$

where $\alpha < 1/m$. Rearranging the above equation and using the approximation, $\ln\left(1-\frac{m_c+1}{m}\right) \approx -\frac{m_c+1}{m}$, we get

$$n_x > c_{mx}\frac{m}{m_c+1}, \tag{11}$$

where $c_{mx} = -\ln\alpha$ is a constant.

The number of mixing events required to climb one step of the ladder is therefore proportional to the number of building blocks. Similar to the mixing probability, we assume n_x to be constant with respect to the mixing level:

$$n_x \approx n_x\left(m_c = 1\right) = \frac{1}{2}c_{mx}m. \tag{12}$$

This approximation is slightly conservative, but bounds the mixing behavior quite accurately as will become clear in the following paragraphs. Recognizing

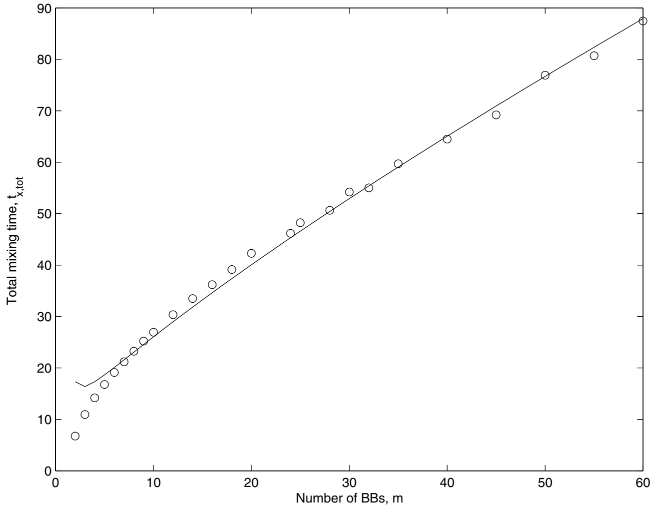

Fig. 5. Comparison of total mixing time, $t_{x,\text{tot}}$ predicted by Eq. (14) with empirical results. The total mixing time is plotted as a function of number of BBs. The empirical results are for 4-bit trap function and are averaged over 500 independent runs

that in a generation, we have $\frac{n}{2}p_c$ recombinations, and using Eqs. (9) and (12), we can calculate the mixing time, t_x:

$$t_x = \frac{n_x}{\frac{n}{2}p_c p_{\text{mix}}} = c_{x,m}\frac{m}{np_c}\left[\frac{(m-1)\left((m-2)k+3\right)}{m(mk-1)}\right], \tag{13}$$

where $c_{x,m} = \frac{2}{3}c_{mx}$ is a constant. The above mixing time refers to the time – in number of generations – required to climb one step of the mixing ladder. However, we are interested in the total mixing time, $t_{x,\text{tot}}$, that is, the time required to go from mixing level 1 to mixing level m. Note that this is the time required to climb $m-1$ steps of the mixing ladder. Approximating, $m-1 \approx m$, we get

$$t_{x,\text{tot}} = mt_x = c_{x,m}\frac{m^2}{np_c}. \tag{14}$$

The model suggests that the total mixing time grows quadratically with the number of building blocks and is inversely proportional to the population size and crossover probability. The mixing-time model (Eq. (14)) is compared to empirical results in Fig. 5, where the total mixing time is plotted as function of number of BBs, m. The results indicate that the agreement between theoretical and experimental results gets better with the number of BBs.

In the analysis presented above, we have assumed that every individual in the population contains either a BB (all ones) or a deceptive attractor (all zeros). However, in the actual case, the probability of having a BB in a string is 2^{-k}.

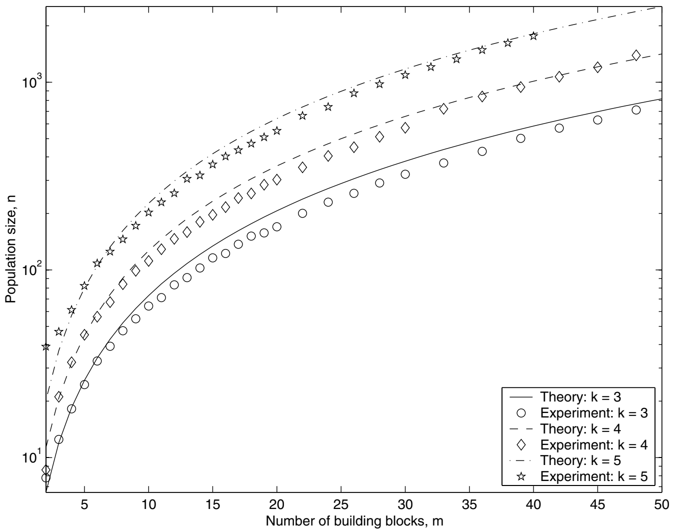

Fig. 6. Empirical validation of the population-sizing model dictated by BB mixing (Eq. (15)). Minimum population size required for a GA success is plotted as a function of number of BBs m. The minimum population size is determined by a bisection method and the results are averaged over 25 independent bisection runs

Therefore, the proportion of recombinations that can result in a mixing event is given by $2^{-mk}\left(2^{(m-1)k}-1\right) \approx 2^k$. Multiplying this term as a with the mixing-time model, yields us the total mixing time when 2^k schemata are present.

As mentioned earlier, to ensure innovative success, we need $t_x < t_{\text{conv}}$. From Eqs. (14) and (4), we get

$$n > c_x \frac{I}{p_c} 2^k m \sqrt{\frac{m}{k}} \tag{15}$$

where $c_x = c_{x,m}/c_c$. Equation (15) suggests that the population size grows as $O(2^k m\sqrt{m})$ as opposed $O(2^k\sqrt{m})$, predicted by the gambler's ruin model [26].

The BB-mixing based population-sizing model (Eq. (15)) is verified with empirical results in Fig. 6. The figure plots the minimum population size required to obtain a solution of desired quality as a function of number of BBs. Population sizes for three different BB sizes, $k = 3, 4$, and 5 are compared. The empirical results are obtained as described in Sect. 4. The figure shows that the empirical results show good agreement with the model prediction.

These results not only demonstrate the utility of building facetwise models, but also brings forth two key behavior of one-point crossover:

1. BB supply and decision making [26] bounds the population size for small problems, and BB mixing governs population sizing for larger problems.

2. The range – in terms of number of BBs – in which BB decision-making and supply bound population sizing is dependent on the BB size. The range increases with the BB size.

Therefore, there is a boundary which segregates the problem space into two regions. One where BB decision-making governs the population sizing, and the other where BB-mixing governs the population sizing.

6 Summary and Conclusions

This study analyzed the mixing behavior of single-point crossover on boundedly decomposable additive fitness functions with tight linkage. Models for predicting the time required for achieving innovative mixing – first for a two building block problem, and then for a general m building block case – are developed. Two key features of single-point crossover, namely the *ladder-climbing* phenomenon, and the *length-reduction* phenomenon are observed empirically and successfully used in developing the mixing time model for m-BB case.

The resulting mixing time models are compared with existing convergence-time model to investigate scenarios which result in *innovative success*. Using such a comparison, population-sizing model dictated by BB mixing is derived. All the models derived in this study are compared with empirical results. Empirical results show good agreement with theory, thus validating the facetwise models.

Results indicate that the population-sizing required for a GA success is dictated by building-block mixing rather than building-block-wise decision-making or building-block supply for moderate-to-large problems. The minimum population size scales as $O\left(2^k m\sqrt{m}\right)$, as opposed to $O\left(2^k\sqrt{m}\right)$ which is the case for advanced operators that adapt linkage. This suggests that even under the best case scenario – that is, assumption of tight linkage – fixed crossover operators are less efficient in comparison to operators that adapt linkage.

Acknowledgments. This work was sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant F49620-00-0163, the National Science Foundation under grant DMI-9908252, and the CSE fellowship, UIUC. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the National Science Foundation, or the U.S. Government.

We thank Martin Butz, Martin Pelikan, and Dirk Thierens for their comments.

References

1. Holland, J.H.: *Adaptation in natural and artificial systems.* University of Michigan Press, Ann Arbor, MI (1975)
2. Goldberg, D.E.: *Genetic algorithms in search optimization and machine learning.* Addison-Wesley, Reading, MA (1989)
3. Goldberg, D.E., Deb, K., Clark, J.H.: Genetic algorithms, noise, and the sizing of populations. *Complex Systems* **6** (1992) 333–362 (Also IlliGAL Report No. 91010).
4. Goldberg, D.E.: *Design of innovation: Lessons from and for competent genetic algorithms.* Kluwer Academic Publishers, Boston, MA (2002)
5. Goldberg, D.E., Thierens, D., Deb, K.: Toward a better understanding of mixing in genetic algorithms. *Journal of the Society of Instrument and Control Engineers* **32** (1993) 10–16 (Also IlliGAL Report No. 92009).
6. Thierens, D., Goldberg, D.E.: Mixing in genetic algorithms. *Proceedings of the Fifth International Conference On Genetic Algorithms* (1993) 38–45.
7. Sastry, K., Goldberg, D.E.: Analysis of mixing in genetic algorithms: A survey. IlliGAL Report No. 2002012, University of Illinois at Urbana-Champaign, Urbana, IL (2002)
8. De Jong, K.A.: An analysis of the behavior of a class of genetic adaptive systems. PhD thesis, University of Michigan, Ann-Arbor, MI (1975)
9. Syswerda, G.: Uniform crossover in genetic algorithms. *Proceedings of the Third International Conference on Genetic Algorithms* (1989) 2–9.
10. De Jong, K.A., Spears, W.M.: A formal analysis of the role of multi-point crossover in genetic algorithms. *Annals of Mathematics and Artificial Intelligence* **5** (1992) 1–26.
11. Goldberg, D.E., Sastry, K.: A practical schema theorem for genetic algorithm design and tuning. *Proceedings of the Genetic and Evolutionary Computation Conference* (2001) 328–335 (Also IlliGAL Report No. 2001017).
12. Christiansen, F.B.: The effect of population subdivision on multiple loci without selection. In Feldman, M.W., ed.: *Mathematical Evolutionary Theory.* Princeton University Press, Princeton, NJ (1989) 71–85.
13. Booker, L.B.: Recombination distributions for genetic algorithms. *Foundations of Genetic Algorithms* **2** (1993) 29–44.
14. Eshelman, L.J., Caruana, R.A., Schaffer, J.D.: Biases in the crossover landscape. *Proceedings of the Third International Conference on Genetic Algorithms* (1989) 10–19.
15. Eshelman, L.J., Schaffer, J.D.: Productive recombination and propagating and preserving schemata. *Foundations of Genetic Algorithms* **3** (1995) 299–313.
16. Rabani, Y., Rabinovich, Y., Sinclair, A.: A computational view of population genetics. *Random Structures & Algorithms* **12** (1998) 313–334.
17. Bridges, C.L., Goldberg, D.E.: An analysis of reproduction and crossover in binary-coded genetic algorithm. *Proceedings of the Second International Conference on Genetic Algorithms* (1987) 9–13.
18. Nix, A.E., Vose, M.D.: Modeling genetic algorithms with Markov chains. *Annals of Mathematics and Artificial Intelligence* **5** (1992) 79–88.
19. Prügel-Bennett, A., Shapiro, J.L.: An analysis of genetic algorithms using statistical mechanics. *Physical Review Letters* **72** (1994) 1305–1309.
20. Stephens, C., Waelbroeck, H.: Schemata evolution and building blocks. *Evolutionary computation* **7** (1999) 109–124.
21. Thierens, D.: Scalability problems of simple genetic algorithms. *Evolutionary Computation* **7** (1999) 331–352.

22. Deb, K., Goldberg, D.E.: Analyzing deception in trap functions. *Foundations of Genetic Algorithms* **2** (1993) 93–108 (Also IlliGAL Report No. 91009).
23. Mühlenbein, H., Schlierkamp-Voosen, D.: Predictive models for the breeder genetic algorithm: I. continuous parameter optimization. *Evolutionary Computation* **1** (1993) 25–49.
24. Bäck, T.: Selective pressure in evolutionary algorithms: A characterization of selection mechanisms. *Proceedings of the First IEEE Conference on Evolutionary Computation* (1994) 57–62.
25. Thierens, D., Goldberg, D.E.: Convergence models of genetic algorithm selection schemes. *Parallel Problem Solving from Nature* **3** (1994) 116–121.
26. Harik, G., Cantú-Paz, E., Goldberg, D.E., Miller, B.L.: The gambler's ruin problem, genetic algorithms, and the sizing of populations. Evolutionary Computation **7** (1999) 231–253 (Also IlliGAL Report No. 96004).