On the Practical Genetic Algorithms

Chang Wook Ahn
Samsung Advanced Institute of Technology
Giheung-Eup, Gyeonggi-Do 449-712, Korea
cwan@evolution.re.kr

Sanghoun Oh, R. S. Ramakrishna
Gwangju Institute of Science and Technology
Oryong-dong, Buk-gu, Gwangju 500-712, Korea
{oosshoun,rsr}@gist.ac.kr

ABSTRACT

This paper offers practical design-guidelines for developing efficient genetic algorithms (GAs) to successfully solve real-world problems. As an important design component, a practical population-sizing model is presented and verified.

Categories and Subject Descriptors: Computing Methodologies [Artificial Intelligence]: Heuristic Methods.

General Terms: Algorithms.

Keywords: Genetic algorithms, practical design-guidelines.

1. INTRODUCTION

Over the last decade, genetic algorithms (GAs) have been successfully applied to problems in business, engineering, and science [1, 2, 3]. This is a consequence of a noteworthy progress in their theory, design and development. On the basis of innovation intuition, in special, a design-decomposition theory has been proposed for developing *competent* (selectorecombinative) GAs, which are a class of GAs that solve hard problems quickly, accurately, and reliably [4]. The design decomposition consists of seven steps: 1) Know what GAs process – building blocks (BBs) 2) Know thy BB challengers – BB-wise difficult problems 3) Ensure an adequate supply of raw BBs 4) Ensure increased market share for superior BBs 5) Know BB takeover and convergence times 6) Make decisions well among competing BBs 7) Mix BBs well [4, 5]. In spite of considerable work on various aspects of GAs, practitioners often face hurdles in confronting real-world problems due to unavailability of problem dependent information. The purposed of this paper is to fill the long standing gap between theory and practices of GAs.

2. PRACTICAL DESIGN GUIDELINES

There are six issues that lead to practical GA design.

1. Representation: This issue is primarily related to the encoding scheme. It is hard to find an encoding method (i.e., representation) that transforms a problem so as to reduce or preserve the intrinsic difficulty of the problem. Hence, the encoding method that has identical genotype and phenotype (of the decision variables) is advisable. Although fixed-length individuals are generally desirable, their variability is not a critical factor provided their design is easy.

2. Initialization: In general, there are two issues to be considered for population initialization: the initial population size and the procedure to initialize the population. At first, the initial population size connected to the supply of raw BBs is crucial for efficiency of GAs in terms of both optimality and complexity. Secondly, there are two ways to generate the initial population: *random* and *heuristic*. If no prior information on the problem is available, random initialization is the natural choice; otherwise, heuristic initialization is favored. Although the mean fitness of the heuristic initialization is already high so that it may help the GAs to find solutions faster, it may just explore a small part of the solution space and never find global optimal solutions because of lack of diversity in the population. In the heuristic case, thus, a portion of the population can still be generated randomly to ensure some diversity in the population. It is noted that the random initialization is generally desirable for stability and simplicity of GAs even when a valuable piece of information is available.

3. Fitness function: The fitness function interprets the individual in terms of physical representation and evaluates its *fitness* based on desired traits (in the solution). It is suggested that the fitness function fully reflect the physical objective of the problem.

4. Genetic operators: The genetic operators must be carefully designed as they directly affect the performance of GAs. *Selection* focuses on the exploration of promising regions in the solution space. As proportionate selection is very sensitive to the selection pressure, a scaling function is employed for redistributing the fitness range of the population. The selection pressure of the ordinal selection is independent of the fitness distribution, and is based solely based on the relative ranking of the population although it may also suffer from high selection pressure. In general, the ordinal selection is preferable. Among the selection schemes (in the ordinal selection), tournament selection without replacement is advisable due to its capability of achieving low (selection) noise. *Crossover* is the primary operator that increases the exploratory power of GAs. In order to successfully achieve the cross-fertilizing type of innovation, crossover operator must ideally intermix good subsolutions without any disruption of the partitions. In practice, uniform crossover is pessimistic as most of real-world problems have the decision variables that are closely interacted each other. Moreover, (population) building-block crossover may also be undesirable because the capability of learning linkage is an essential prerequisite of the operator. In stead of pursuing the maximum BB-wise mixing in the population, it can be efficient to increase the population size and employ a simple crossover that has a low probability of dis-

GECCO'05, June 25–29, 2005, Washington, DC, USA.
ACM 1-59593-010-8/05/0006.

rupting the BBs found so far. Therefore, it is recommended that building-block crossover is suitable if the evaluation of fitness function requires a high computational cost; otherwise, one- or two-point crossover is desirable. Naturally, the crossover probability must be relatively high. *Mutation* is the secondary operator of GAs to explore a solution space. To carry out the continual improvement type of innovation, as in nature, the probability of applying mutation must be very low. Hence, the suggestion is that any type of mutation designed is applicable as long as its probability is not very high. Moreover, it is possible to get rid of mutation when the design of mutation operator is complicated.

5. Treating infeasible individuals: In case that a problem has some constraints, crossover or mutation may often generate infeasible individuals that violate the constraints. There are two strategies to deal with infeasible individuals: one is to impose a penalty and the other is to repair them. Although a classical method employs penalty functions, it is not easy to come up with an appropriate penalty function. Moreover, this technique may sacrifice some feasible individuals as well because the infeasible individuals might continue to be reproduced. On the other hand, the repair method is applied extensively. But it is not always simple to cure infeasible individuals. Hence, the repair strategy is always advisable unless developing a repair function is an arduous task or the designed function is computationally too expensive by far.

6. Population size: A problem that arises with GAs is to properly estimate the values of parameters. Most of the parameters can be determined by the transcendental cognition of practitioners so as to attain good performance. However, it is not easy to estimate the population size that guarantees an optimal solution quickly enough. Thus, the population size has generally been perceived as the most important factor. With this in view, the population-sizing model [1] (described in Section 3) is recommended for the practical use.

3. PRACTICAL POPULATION SIZE

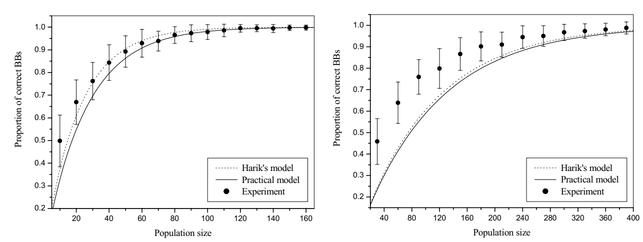
A fancy study has developed a refined population-sizing model by integrating the requirements of the BB supply and decision making [5]. It provides an accurate bound on determining an adequate population size that guarantees a solution with desired quality for (selectorecombinative) GAs. However, it requires stochastic information such as the variance of fitness (i.e., noise variance σ_{bb}) and the expected difference value of fitness (i.e., signal d) between the best and the second-best BBs, which may not be available in many practical problems.

Assume that individuals consist of m non-overlapping (i.e., separable) and uniformly scaled BBs of size k. The assumptions follow from the decomposition even when the problem is not separable. It is also assumed that all the competing BBs are ordered and they are uniformly distributed. The assumptions are valid when ordinal selection is employed for real-world problems [1]. Note that standard deviation can be thought of as the probabilistic "width" or "spread" of distribution of a random variable; indeed, the factor $2\sigma_{bb}$ represents the total average range of fitness changes of all the BBs. Thus, the signal d can be represented as $\frac{2}{\chi^k-1}\sigma_{bb}$, where χ is the (average) cardinality of the alphabet. Employing the signal d makes Harik's decision model practically available; it leads to a practical decision model. Note that a GA succeeds when all the N members of the population in the BBs of interest are correct. It can be modeled by the gambler's ruin process [5]. From the initial BB supply, practical decision model, and the gambler's ruin process, a practical population-sizing model can be obtained as [1]

$$N = -\frac{\chi^k}{2}\ln(\alpha)\left(\frac{\chi^k-1}{2}\sqrt{\pi\left(\frac{n}{k}-1\right)}+1\right).$$

Here, n and α are the length of individuals and the probability of GA failure.

The model is verified with test problems; an one-max problem, a minimal deceptive problem (MDP) [3], and a modified 3-bit deceptive problem [2]. (The modification is to fulfill the assumptions.) With regard to the one-max problem, the practical model is equivalent to Harik's model; thus, its validity is clearly supported. The results for deceptive problems are shown in Fig. 1. (One-point crossover is used for avoiding the excessive disruption of BBs [5].) It is observed that the analytical model is consistent with the experimental results even for higher population size. Moreover, the close agreement between the practical population-sizing model and Harik's model implies that the pratical decision model can accurately approximate the actual SNR without any statistical information about the competing BBs.



(a) Results for the MDP (b) 3-bit deceptive problem

Figure 1: Verification of the population-sizing model

4. CONCLUSION

Design of practical GAs for solving real-world problems was the main focus all along. Further, this paper also investigated a practical population-sizing model that comes in handy in determining an adequate population size for finding a desired solution without requiring statistical information such as the signal or variance of competing BBs.

5. REFERENCES

[1] C. W. Ahn and R. S. Ramakrishna, "A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations," *IEEE Transactions on Evolutionary Computation*, 6(6):566–579, 2002.

[2] C. W. Ahn, *Theory, Design, and Application of Efficient Genetic and Evolutionary Algorithms*, Doctoral Dissertation, Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, 2005.

[3] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, MA: Addison-Wesley, 1989.

[4] D. E. Goldberg, *The design of innovation: Lessons from and for competent genetic algorithms*, Kluwer Academic Publishers, 2002.

[5] G. Harik, E. Cantú-Paz, D. E. Goldberg, and B. L. Miller, "The Gambler's Ruin Problem, Genetic Algorithms, and the Sizing of Populations," *Evolutionary Computation*, 7(3):231-253, 1999.