

Improving Small Population Performance under Noise with Viral Infection + Tropism

Yuji Sato
Dept. of Comp. and Info. Sci.
Hosei University
Tokyo 184-8584 JAPAN
+81-42-387-4533
yuji@k.hosei.ac.jp

David E. Goldberg
Dept. of Entrepreneurial Eng.
Univ. of Illinois at Urbana-Champaign
Urbana, IL 61801 USA
+1-217-333-0897
deg@uiuc.edu

Kumara Sastry
Dept. of Entrepreneurial Eng.
Univ. of Illinois at Urbana-Champaign
Urbana, IL 61801 USA
+1-217-333-2346
ksastry@uiuc.edu

ABSTRACT

In this paper we report on the effect of viral infection with tropism on the formation of building blocks in genetic operations. In previous research, we applied genetic algorithms to the analysis of time-series signals with noise. We demonstrated the possibility of reducing the number of required entities and improving the rate of convergence when searching for a solution by having some of the host chromosomes harbor viruses with a tropism function. Here, we simulate problems having both multimodality and deceptiveness features and problems that include noise as test functions, and show that viral infection with tropism can increase the proportion of building blocks in the population when it cannot be assumed that a necessary and sufficient number of entities are available to find a solution. We show that this capability is especially noticeable in problems that include noise.

Categories and Subject Descriptors

D.3.3 [Programming Languages]: General

General Terms: Design, Experimentation, Verification.

Keywords: Genetic Algorithms, Virus Infection, Tropism, Building Block, Deceptive Problem, Multimodality, Noisy Optimization.

1. INTRODUCTION

The importance of building blocks (BBs) and their role in the workings of genetic algorithms (GAs) have been reported, and the following six conditions for a GA success have been proposed (Goldberg, Deb, & Clark, 1992): (1) Know what GAs are processing – building blocks (BBs), (2) ensure an adequate initial supply of raw BBs, (3) ensure growth of superior BBs, (4) ensure the mixing of BBs, (5) ensure good decisions among competing BBs, and (6) solve problems with bounded BB difficulty. One of the important conditions is to make sure that GA is well supplied with a sufficient supply of the BBs required to solve a given problem (Goldberg, Sastry, & Latoza, 2001). It is also important that the proportion of the good BBs in the population grow.

On the other hand, GA can be used to solve hard problems such as deceptive problems, multimodal problems, and the optimal analysis of noisy data. For example, it is reported that the messy GA (Goldberg, Korb, & Deb, 1989) and the Bayesian

optimization algorithm (BOA) (Pelikan, Goldberg, & Cantu-Paz, 1999) is effective for a deceptive problem. Niching in GA and the distributed GA (Tanese, 1989) is effective for multimodality. A sufficient number of population and GA with virus are effective for the optimal analysis of noisy data. GA with viral tropism has a possibility to improve the convergence rate of solution searches in genetic algorithms with parasitic viruses and to reductions in the number of host entities (Sato, Yasuda, & Goto, 2007).

In this paper we report on the results obtained when attempting to improve the proportion of the good BBs in the population with respect to trap functions with multimodality by grafting viruses with tropism on to part of the chromosomes used for simple GA. Trap functions are known to be hard for simple GAs to solve.

2. NICHING AND VIRAL TROPISM

The basic idea behind viral infection is that the growth of building blocks (schema associated with high scores) can be assisted in chromosomes acting as host entities by the exchange of gene locus information and genetic information possessed by viruses among the host entities. At the same time, there is the fear that a highly infectious virus could have an adverse effect on a host to the point that it becomes extinct, or that high infectiousness could promote convergence to a local solution. As one means of solving this problem, we proposed a technique that controls viral infection based on the concept of tropism.

Figure 1 shows a conceptual diagram depicting the effect of introducing tropism in GA having a viral infection function. First, with respect to standard GA, the entire population can be divided into a number of subpopulations. A GA can then be applied to each of these subpopulations and data exchanged among them at appropriate times in a technique called “distributed GA.” This technique has been reported to be particularly effective in multimodal problems (Tanese, 1989). There is also the “GA with virus” technique that aims to increase the proportion of building blocks in the population by propagating useful schema to other hosts via viruses. Here, the introduction of a tropism function in GA with viral infection enables infection to occur only between a host and virus having similar schema. Tropism can therefore be treated as one way of introducing a niche function into viral infection, and the use of tropism is considered to enhance GA performance in multimodal problems. Niching is a method by which multiple “species” are preserved in different “niches” within the problem domain through modifications to the selection procedure that values useful diversity. When niches are introduced and sized properly, effective diversity can exist for a very long time (Goldberg, 2002).

In this regard, GA with viral infection may be thought of as a form of distributed GA, but it has several differences with standard distributed GA. First, in distributed GA, the designer must decide beforehand how to go about dividing the population and how many subpopulations to create. In the case of tropism, however, groups formed by the correspondence between viruses and hosts targeted for infection are automatically determined by the definition of the tropism in question. A certain host entity may belong to multiple groups (may be infected by multiple viruses) or may not even be the target of viral infection. Next, a subpopulation, once decided upon in distributed GA, is basically fixed, while with tropism, the correspondence between viruses and hosts changes in a dynamic fashion from one generation to the next. Finally, while the population for performing GA is itself divided into subpopulations in distributed GA, there is only one population to apply GA to when introducing tropism in GA with viral infection. With tropism, groups are formed indirectly via viruses in terms of a correspondence between viruses and hosts.

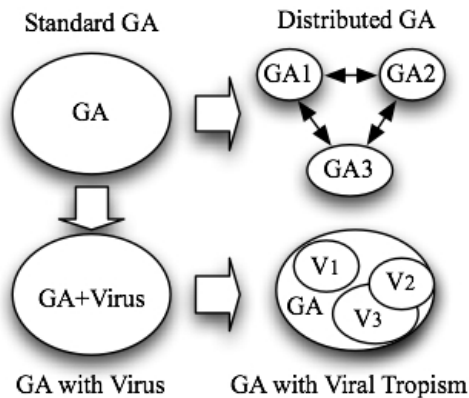


Fig. 1. Conceptual diagram depicting the effect of introducing tropism in GA with virus.

3. EVALUATION AND DISCUSSIONS

We use 8 copies of a 4-bit trap function and this function with noise as a test function. For crossover rate $p_c=1.0$ and mutation rate $p_m=0.0$ show that all 8 optimal solutions can be obtained by SGA if a necessary and sufficient number of entities can be obtained for searching. In this case, the benefits of adding viruses or a tropism function could not be observed. If, however, a necessary and sufficient number of entities cannot be obtained, the addition of a viral-infection function to SGA as well as the addition of a tropism function can increase the proportion of building blocks in the population and obtain more optimal solutions. For example, for the case of 20 entities and the 8-copies 4-bit trap function, not even one of the 8 optimal solutions could be found by SGA, but when adding viral infection, 2 optimal solutions could be found, and when adding the tropism function, 4 optimal solutions could be obtained. When resources and time are limited, as is the case with any real-world problem, it is imperative to use those resources judiciously. Therefore it is very important to improve performance at a small population size. The reason for this improvement in search performance by the addition of viral infection is thought to be that effective schemas propagate by viruses to other host entities thereby increasing the proportion of correct building blocks in the population. Also, the reason for an even better score by adding a tropism function is considered to

be that the efficiency of viral infection increases with tropism, or that the niche function takes effect when adding tropism resulting in greater multimodality search performance. Nevertheless, there was no effect to the point of detecting all optimal solutions here even when adding a virus or tropism function. It is that viral tropism improves the population sizing required for good performance marginally, but it was not enough to solve the GA hard problem. To find the reason for this, we examined the relations between number of generations and maximum score, average score, and the enabling/disabling of viral infection. We found that host entities converge to a local solution before the proportion of building blocks can be sufficiently increased by the addition of a virus or tropism function. To enable all optimal solutions to be detected, future research must investigate ways of delaying convergence to local solutions by SGA and of increasing the speed of building-block generation by viruses and tropism.

For the case that an appropriate mutation rate could be set, it was found that all 8 optimal solutions to an 8-copies 4-bit trap function could be obtained by SGA even for relatively small population. No particular benefits of adding a virus or tropism function could be observed here. When adding noise to the trap function, however, it was found that adding viral infection as well as a tropism function to SGA could increase the proportion of building blocks in the population and obtain a greater number of optimal solutions. This effect was even more profound when a small population size. From evaluations we can know that it takes more generations for all host entities converge to the local solutions under the noise in the evaluation function and viral tropism could take a time to generate new good building blocks. It can therefore be considered that the addition of a virus or tropism function has the effect of improving search performance with respect to noise.

4. REFERENCES

- [1] Goldberg, D. E. (2002). *The Design of Innovation*. Kluwer Academic Publishers.
- [2] Goldberg, D. E., Deb, K., & Clark, J. (1992). Genetic algorithms, noise, and the sizing of populations. *Complex Systems*, 6, 333-362.
- [3] Goldberg, D. E., Korb, B., and Deb, K. (1989). Messy genetic algorithms: Motivation, analysis, and first results. *Complex Systems*, 3(5), 493-530.
- [4] Goldberg, D. E., Sastry, K., and Latoza, T. (2001). On the supply of building blocks. In *Proceedings of the 2001 Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers, 336-342.
- [5] Pelikan, M., Goldberg, D. E., and Cantu-Paz, E. (1999). BOA: The Bayesian optimization algorithm. In *Proceedings of the 1999 Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers, 525-532.
- [6] Sato, Y., Akatsuka, Y., and Nishizono, T. Reward Allotment in an Event-driven Hybrid Learning Classifier System for Online Soccer Games. In *Proceedings of the 2006 Genetic and Evolutionary Computation Conference*, ACM Press, Seattle, 2006, pp. 1753-1760.
- [7] Tanese, R. (1989). Distributed Genetic Algorithms. In *Proceedings of the Third International Conference on Genetic Algorithms*, 434-439.