Guided Mutations in Cooperative Coevolutionary Algorithms for Function Optimization

Chun-Kit Au
Flat F, 22 / F, Block 3,
Kenswood Court, Kingswood Villas,
Tin Shui Wai, Hong Kong, China
vincentau@alumni.cuhk.net

Ho-Fung Leung
Department of Computer Science and Engineering,
The Chinese University of Hong Kong,
Hong Kong, China
lhf@cse.cuhk.edu.hk

ABSTRACT
In this study, a mutation method called "guided mutation" is proposed. In guided mutation, each individual maintains two mutation control parameters: mutation direction and mutation step size. Guidance in mutations is provided by continuous updates on these parameters during the evolutionary process. We apply guided mutations to cooperative coevolutionary algorithm (CCEA), and compare its performance of optimizing nine common problem domains with those of other CCEAs. Our results show that guided mutations can improve the performance of a CCEA in some problem domains. We discuss the implications of our results and suggest some directions for future research.

Categories and Subject Descriptors
I.2.m [**Miscellaneous**]: Evolutionary Computation

General Terms
Algorithms, Experimentation, Performance

Keywords
Cooperative Coevolution, Guided Mutation

1. INTRODUCTION
In cooperative coevolutionary algorithm (CCEA) [1], a problem is divided into modular sub-components. Each species is assigned one sub-component, and evolves separately. A complete solution is assembled by choosing individuals from each species and combining them into a solution. In other words, individuals of each species collaborate to solve the problem. Fitness values of the individuals are evaluated based on how well that complete solution can solve the problem. The better their collaborations, the higher their fitness values when they are evaluated.

2. GUIDED MUTATIONS
Like evolutionary algorithms, CCEAs search their potential solutions in their search space. Variations of individuals are by genetic operations such as crossover and mutations. These genetic operations are stochastic processes that randomly explore the search space. No information is given on guiding these operations to explore the search space. To explore the search space in a effective and efficient way, we believe that genetic operations should be guided by taking into account of searching directions and searching step sizes. In this study, we propose a method to guide mutations in CCEAs.

In guided mutations, each individual is assigned its own mutation direction and mutations step size. Mutation direction is the searching direction in the search space when an individual is mutated. Mutation step size is the step size in the search space moved by a mutation operation. Both of them control how an individual is mutated and are continuously updated during evolution based on the criteria: as long as that mutation direction is still beneficial to the individual, its mutation will keep the tendency to search that direction. Once that mutation direction is no longer advantageous, the mutation direction will then be reversed. In addition, if the mutation direction remains unchanged, the mutation step size is increased. If the mutation direction is reversed, the mutation step size is decreased. After evaluating the changes on these parameters (*i.e.* whether keeping the same mutation direction or reversing to the opposite mutation direction, and whether increasing or decreasing the mutation step sizes), an individual can be mutated according to its own mutation direction and mutation step size.

3. DISCUSSIONS AND FUTURE WORKS
We apply guided mutations to Potter's CCEA model [1]. Nine problem domains of function optimization are chosen from the current literature. Experiments are conducted to compare the performance of a CCEA adopting guided mutations with those of other CCEAs. Our results show that guided mutations can improve the performance of a CCEA in some problems domains. This provides a promising area of guiding mutations in cooperative coevolution for function optimization.

There are several directions for future research. One future direction is to explore other methods of guiding the mutation. Our current approach requires additional fitness evaluations to evaluate the changes on the two mutation control parameters, and it is not desirable when fitness evaluations cost expensive computations. Another direction is to study the relationship between the problem structure and the guided mutation, since we believe that guidance may not perform well in some problem domains, based on our observations on the experimental results. Lastly, we can apply guidance in crossover operations and investigate its effects on the performance of a CCEA.

4. REFERENCES
[1] M. Potter and K. De Jong. A cooperative coevolutionary approach to function optimization. In *Third Conference on Parallel Problem Solving from Nature*. Springer, 1994

GECCO'07, July 7–11, 2007, London, England, United Kingdom
ACM 978-1-59593-697-4/07/0007.