JGBL paradigm: A Novel Strategy to Enhance the Exploration Ability of NSGA-II

Ke Li
Department of Computer Science
City University of Hong Kong
Hong Kong, China
keli.genius@gmail.com

Sam Kwong
Department of Computer Science
City University of Hong Kong
Hong Kong, China
cssmak@cityu.edu.hk

Kim-Fung Man
Department of Electronic Engineering
City University of Hong Kong
Hong Kong, China
eekman@cityu.edu.hk

ABSTRACT

NSGA-II is one of the most efficient multi-objective evolutionary algorithms (MOEAs) for solving multi-objective optimization problems (MOPs). In this paper, a Jumping Gene Based Learning (JGBL) paradigm is proposed to enhance the exploration ability of NSGA-II. JGBL paradigm simulates the natural behavior of maize and is incorporated into the framework of the original NSGA-II. It only operates on the non-dominated solutions which are eliminated in the environmental selection procedure due to the low quality of crowded distance. The activation of JGBL operation is entirely adapted online according to the search status of evolutionary process to give a needed fuel when the population evolves slowly with inherent variation operators.

Categories and Subject Descriptors

I.2.8 [[**Computing Methodologies**]: Artificial Intelligence—*Problem Solving, Control Methods, and Search*

General Terms

Algorithms, Performance, Reliability.

Keywords

Multi-objective optimization, jumping gene based learning, NSGA-II

1. INTRODUCTION

Exploration and exploitation are two important parts of the search dynamics of multi-objective evolutionary algorithms (MOEAs). Generally speaking, exploitation corresponds to the mechanism that related to the selection of high fitness individuals from the current population, while exploration represents the function that concerning with the generation of offspring from parental population by means of genetic variation. However, most of the existing studies about MOEAs mainly focus on fitness assignment, diversity preservation, or environmental selection, despite the equal importance of exploitation and exploration contributing to the performance of MOEAs. On the contrary, the exploration part commonly follows traditional routine of genetic variation such as crossover and mutation.

In this paper, our contribution is proposing Jumping Gene Based Learning (JGBL) paradigm to acclimate to the framework of the state-of-the-art NSGA-II, in order to enhance its exploration ability. The original jumping gene paradigm have been proposed in our previous work about JGGA [**?**][**?**]. The differences between our JGBL paradigm and original JGGA are two folds. On the one hand, the jumping gene operators in JGBL paradigm are completely adapt to the continuous search space. On the other hand, JGBL paradigm is no longer playing as an alternative variation operator which operates on the entire population. The fundamental motivation is to explore the non-dominated individuals which are eliminated by the environmental selection procedure of NSGA-II, so as to inject needed energy when the inherent genetic variations meet some difficulties to propel the population forward.

2. OUR PROPOSED APPROACH

The mechanisms of transposons, namely *cut and paste* and *copy and paste*, are originally operated on the DNA sequences. Thus, a binary coded chromosome would be a straightforward way to mimic these behaviors within the framework of EA. As for continuous search space, instead of discrete "0-1" binary strings, an individual is usually made up of several real variables. Nevertheless, an individual could also be represented by a chromosome whose genes are encoded by these variables, without losing any generality.

2.1 Remedy Strategy For Boundary Violation

As discussed in [**?**], arbitrary switches of the information contained in genes are usually not permitted due to the boundary violation problem. Hence, some kind of remedy techniques should be considered in order to keep the variables in valid ranges after information exchange.

Definition 1. The percentage of a real number that occupies in its range is defined as the variable information of it maintained in this range. It could be calculated as the following formulation:

$$Information(x) = \frac{x - lower(x)}{upper(x) - lower(x)}$$

where $Information(x)$ means the variable information maintained by x, and $upper(x)$ and $lower(x)$ represent the upper and lower bounds of the range that x resides in, respectively.

In order to solve the problem caused by boundary violation, we assume that the variable information of a variable

in its original position would be remained, no matter where it is switched to. Thus, the value after it transfer to a new position could be restored as the following formulation:

$Transfer(x) = lower(y) + Information(x) \times [upper(y) - lower(y)]$

where $Transfer(x)$ indicates the value after x is transferred to a new gene position y, while $lower(y)$ and $upper(y)$ are the lower and upper bounds of the range in the gene position y, respectively.

2.2 Computational Formation of Jumping Gene Operator

As introduced in the previous paragraph, the operation of jumping gene is basically upon the mechanism of transposon. In general, transposon is selected from the genes of one chromosome in a random manner. The components of a transposon could be made up by more than one gene. And for a particular chromosome, there is also no specific limitation on the number of transposons. Moreover, the settle positions of the transposons are also chosen randomly. And the contents of the transposons transferred in a horizontal way, which is a type of lateral movement of genes within the same chromosome or even to another individual in the population pool. Transposons themselves are assigned to one of two classes according to their mechanism of transposition, which can be described as either *cut and paste* or *copy and paste*. The implementation of the *cut and paste* operation is that the transposon is cut from the original position and insert into a new site. In the case of *copy and paste* operation, the selected genes replicate themselves to form the transposon, and then this copy is used to replace the genes in the selected positions of the target chromosome.

2.3 JGBL PARADIGM FOR NSGA-II

Here we propose to incorporate the JGBL paradigm into the framework of NSGA-II, in order to enhance the exploration ability of NSGA-II to better approach and spread along the Pareto optimal front. Specifically, JGBL operations are activated only when the cardinality of the non-dominated population is larger than that of the parent population. After the environmental selection procedure of NSGA-II, some of the non-dominated solutions are eliminated due to the poor performance on crowding distance metric. Nonetheless, the JGBL paradigm is just operating on these moderate solutions. At first, a combined population $R_t = P_t \cup Q_t$ is formed, where P_t is the parent population and Q_t is the child population. The size of R_t is $2N$. Then the environmental selection procedure is called to sort R_t according to non-domination. We assume that the number of non-dominated solutions is larger than N at the current generation t. That is to say, the size of the best non-dominated set F_1 surpasses N. In order to choose exactly N population members to form the next generation parent population, NSGA-II sorts the solutions in F_1 according the crowding distance metric in descending order to fill all population slots. Instead of eliminating the solutions that have small crowding distance values in NSGA-II directly, JGBL paradigm aims at mining some useful information through exploring this kind of solutions. Two advantages encourage the application of JGBL paradigm in this way. On the one hand, this kind of solutions possess satisfactory convergence property since no other solutions dominate them in the hybrid population R_t. Applying JGBL operators act on them might have more

chance to generate better mutants compared to those dominated ones. On the other hand, they are eliminated during the environmental selection procedure due to the poor performance on crowding distance metric. In other words, they inhabit in crowded areas. The JGBL paradigm could be also regarded as a local searcher which might be helpful to explore some unknown region in a sense. Thus the overall diversity of population might be enhanced at last by the addition of some favorable solutions.

Similar to the behavior of genetic operations, the operation of JGBL paradigm is also based on the opportunity. However, different from the traditional genetic operators, the activation of JGBL paradigm is not only decided by a predefined probability, but also the status of the current evolution. It is worth noting that all four different jumping gene operators are utilized whenever the JGBL paradigm is activated. In this way, a solution would be able to generate six possible mutants in one time, so that sufficient opportunity could be given to this solution to explore some unknown regions.

3. CONCLUSION

This paper provided a strategy, JGBL paradigm, which could be embedded into the framework of NSGA-II. JGBL paradigm is inspired from the mechanism of transposon in maize. Different from our previous work on jumping gene paradigm, JGBL paradigm is completely designed for solving problems in continuous search space. Furthermore, instead of alternating the state-of-the-art variation operators, the underlying motivation of JGBL paradigm is to explore some helpful information from those moderate solutions, which are non-dominated in current population, but eliminated by the environmental selection due to their poor performance on maintaining uniformity and diversity. In this way, needed energy could be injected to the search engine whenever the evolution gets trapped.