Self-emergence of Structures in Gene Expression Programming

Xin Li

Department of Computer Science, University of Illinois at Chicago, Chicago, IL, 60607
Email: xli1@cs.uic.edu

Abstract

This thesis work aims at improving the problem solving ability of the Gene Expression Programming (GEP) algorithm to fulfill complex data mining tasks by preserving and utilizing the self-emergence of structures during its evolutionary process. The main contributions include the investigation of the constant creation techniques for promoting good functional structures emergent in the evolution, analysis of the limitation with the current implementation scheme of GEP, and introduction of a novel utilization of the emergent structures to achieve a flexible search process for solutions at a higher level.

Problem Statement

First introduced by Candida Ferreira (Ferreira 2001), GEP is a recently developed evolutionary computation method for data analysis and knowledge discovery. Born from Genetic Algorithms (GAs) and Genetic Programming (GP), GEP is both flexible at genetic operations due to its linear genotype and capable of retaining a certain extent of functional complexity due to its phenotype as expression trees. Distinguished from other traditional machine learning methods, GEP searches the global optimum through a population of candidate solutions in parallel and is able to produce solutions of any possible form as computer programs. Previous research work has shown its powerful capabilities over a large range of domains, including regression modeling, classification tasks, parameter optimization, time series prediction, logic synthesis, and so on (Ferreira 2002; Zhou et al. 2003).

However, the algorithm could be improved with respect to both time efficiency and solution quality when dealing with complex problems. These problems are usually featured by large data sets, high dimensional feature sets and non-linear form of hidden knowledge within the data. The proper solutions to these applications are structurally intricate enough to require a more dedicated derivation process. Currently, the exploration of search space is purely driven by the disruptive genetic operations and fitness measure in GEP. However, the structure of a solution, which actually largely determines its functionality, is yet to be explicitly maintained or utilized. We hypothesize that a more effective search of the solution can be built upon the evolving procedure of the solution structures, which we call as *the self-emergence of structures in GEP*. Here *structures* mean physical elements or building blocks that help compose the optimal solution for a problem, which, for example, could be either the overall functional structure of the solution or sub-parts of it (also called *substructures* in the latter case). By saying *self-emergence*, we mean structures are generated, preserved, and evolved automatically during the evolutionary process.

Copyright © 2005, AAAI (www.aaai.org). All rights reserved.

Proposed Plan for Research

Although building blocks, modularity and hierarchy have been historically addressed in GAs and GP (Holland 1975; Koza 1994), our research stands out by accentuating the utilization of naturally emergent structures based on the representation scheme of GEP. We believe only structures adapted to the environment via fitness measurement during the evolution are most competitive candidates for desirable structures of the unknown optimal solution. The proposed research plan gradually digs into the thesis topic as follows:

(1) ***Constant creation methods in GEP***

Since finding the optimal solution structure and the appropriate constant coefficients are intermingled together in the GEP evolutionary process, a discovered good solution structure is less likely inherited if its combination with improper constant coefficients results in a low fitness value, and it may require extra search effort to be rediscovered later. Therefore, given the evolved candidate solution structures at the end of each generation, it might be helpful to incorporate some local search algorithms to optimize constants for speeding up the learning process. This forms the motivation to investigate the constant creation methods in the context of GEP.

As of now, work that has been achieved includes: an extensive survey for existing constant creation methods in the literature about related algorithms; five constant creation methods were defined for GEP based on two basic constant mutation techniques, and all of these methods have been implemented and experimented with several benchmark symbolic regression problems (Li et al. 2004); the experiments have proved the feasibility of specific assistance for GEP in discovering useful numeric constants, which then helps identify good functional structures; it was found that constant creation methods applied to the whole population for selected generations

perform better than methods that are applied only to the best individuals, and have achieved significant improvement in the average fitness of the best solutions.

The initial work shows support for the hypothesis that the GEP evolutionary process propels the emergence of solution structures in its genotype.

(2) ***Analysis of limitations with the implementation scheme of GEP***

Since in GEP, genotype is linear character strings of a fixed length as in GAs, it is possible to analyze the driving force behind the evolutionary process of GEP following the paradigm of the schema theorem from GAs (Holland 1975). This would facilitate us to better understand how the solution structures evolve under the current implementation scheme of GEP, and further propose appropriate techniques to effectively utilize the emergent structures in GEP.

Work has been done includes: a derivation of schema theorem for GEP, which defines the pattern of solutions (i.e., schema) in GEP as a set of gene groups corresponding to the sub-trees in its expression tree representation; based on the schema theorem, it was demonstrated that genetic operators like crossover and rotation, which are designed more in favor of passing genetic material down the generations, are actually as destructive as mutations to the evolved solution structures in GEP; it was also drawn from the analysis that shorter schemata defined by closely related elements in the linear genotype can receive the best exploration of their instances in the later evolutionary process.

The current work shows a more direct mapping between the genotype and phenotype is necessary so that the solution structure encoded in the linear genotype can be better preserved and utilized during evolution.

(3) ***Utilization of emergent structures in GEP***

Motivated by the analysis on the implementation scheme of GEP, it is appealing to preserve and utilize the emergent structures in GEP from simpler components, which will subsequently result in the emergence of higher level solution structures. Moreover, since the final solution is most likely formed by combining gene segments from other candidates, this will ideally help save repeated search effort for discovering useful functional structure components, and yield a more natural way to tackle complex problems by constructing the solution hierarchically within its genotype representation.

Fulfilled work regarding this idea includes: a definition for emergent structural components (called *derived genes*) in GEP, which is basically a set of genes in the genotype corresponding to a complete sub-tree in the phenotype and having a high appearance frequency in the fittest individuals (called the *elite group*) among the whole population (e.g., given "sqrt.*.+.*.a.*.sqrt.a.b.c./.1.-.c.d" as the genotype string of a solution, the substring "-.c.d" forms a sub-tree in the phenotype, and if the appearance frequency requirement is met, it then defines a qualified derived gene); introduction of two novel genetic operators, namely, compression and expansion, to implement derived genes, and as with their application, GEP is able to preserve and utilize the useful emergent structural components completely determined by the evolution itself; implementation of a dynamic substructure library based on the compression and expansion operators in order to incorporate the utilization of emergent structures into the design of GEP algorithm.

The initial experimental results have revealed the promising advantages of this proposed implementation scheme regarding emergent structures.

(4) ***Further plan for this thesis research***

Up to now, the investigation of this thesis topic has basically established the framework. However, many more interesting ideas need further research, and some of them are planned as follows in decreasing order of priority:

• To propose a new genotype-phenotype mapping mechanism so that the linear chromosome can also encode the natural functional hierarchy defined by the expression tree. This would benefit the emergence of the structures in GEP and ease the implementation of the derived genes.

• Further exploration of different kinds of emergent structural components, which generally fall into categories of derived constant, derived attribute and derived function. This can possibly produce a more general and viable scheme to utilize the emergent solution structures.

• To introduce parametric constants into the GEP algorithm. The general idea is that constant genes will participate in the evolution as a symbol. Before the final evaluation of the solutions, the constant symbols will tune up around the original value to maximize the fitness of the current functional structure.

• Extensive benchmark testing on both regression and classification problems, to measure the effectiveness of the new design for the algorithm from different perspectives. The results will be compared to the original GEP, as well as some other traditional methods like neural network and C4.5 decision tree.

References

Ferreira, C. 2001. Gene Expression Programming: a New Adaptive Algorithm for Solving Problems. *Complex Systems* 13(2): 87-129.

Ferreira, C. 2002. *Gene Expression Programming: Mathematical Modeling by an Artificial Intelligence*. Angra do Heroismo, Portugal.

Zhou, C.; Xiao, W.; Nelson, P. C.; and Tirpak, T. M. 2003. Evolving Accurate and Compact Classification Rules with Gene Expression Programming. *IEEE Transactions on Evolutionary Computation* 7(6): 519 – 531.

Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.

Koza, J. 1994. *Genetic Programming II: Automatic Discovery of Reusable Programs*. Cambridge, MA: MIT Press.

Li X.; Zhou, C; Nelson, P. C.; and Tirpak, T. M. 2004. Investigation of Constant Creation Techniques in the Context of Gene Expression Programming. In *Late Breaking Paper at Genetic and Evolutionary Computation Conference (GECCO-2004)*. Seattle, WA, USA.