

Hierarchical Topology/Parameter Evolution in Engineering Design

Kisung Seo

Dept. of Electronics Engineering
Seokyeong University

서기성

서경대학교 전자공학과

E-mail : ksseo@skuniv.ac.kr

Abstract

This paper suggests a control method for efficient topology/parameter evolution in a bond-graph-based GP design framework that automatically synthesizes designs for multi-domain, lumped parameter dynamic systems. We adopt a hierarchical breeding control mechanism with fitness-level-dependent differences to obtain better balancing of topology/parameter search - biased toward topological changes at low fitness levels, and toward parameter changes at high fitness levels. As a testbed for this approach, an eigenvalue assignment problem, which is to find bond graph models exhibiting minimal distance errors from target sets of eigenvalues, was tested and showed improved performance for various sets of eigenvalues.

1. Introduction

Evolutionary computation has been used many times to automate the creation of engineering designs, such as electrical, mechanical, and mechatronic systems. Especially, genetic programming has been used for design of several patented electrical circuits, controllers, and antennas [1,2]. Engineering design of dynamic systems usually involves discovering topological connections of components and optimizing of their numeric parameters, simultaneously, so is topologically open-ended.

The topology specifies the system's structure, which consists of the number and type of components and their interconnections. On the other hand, parameter optimization seeks the best numerical values for given a topology. However, no definitive approach has yet been introduced for topology/parameter evolution.

Most approaches to finding better designs are limited to using huge populations. The key idea of our approach is to provide different breeding probabilities for topology and parameter operations according to the fitness level of each subpopulation in a genetic programming with fitness-stratified populations. Additionally, more topology-altering operations are executed in earlier generations and more parameter-altering operations are executed in later generations.

The Bond Graph / Genetic Programming (BG/GP) design methodology[3] has been developed to overcome limitations of single-domain design approaches and enable open-ended search, based on the combination of these two powerful tools.

As a test class of design problems, we have chosen one in which the objective is to realize a design having a specified set of

eigenvalues. The eigenvalue assignment problem is well defined and has been studied effectively using linear components with constant parameters. Section 2 discusses the nature of the topology/parameter design problem and bond graph synthesis. Section 3 explains hierarchical probability control method. Section 4 presents results for eigenvalue design problems, and Section 5 concludes the paper.

2. Topology/Parameter Design in Bond Graph Synthesis

2.1 The nature of topology/parameter design in engineering problem

Most engineering design problems involve making (or discovering good) topological connections of components and optimizing their numeric values in an open-ended manner. That means both topology and parameter values should be optimized[4].

Topology connection is represented as a directed graph G with vertex set V and edge set E , where $V = \{ v_0, v_1, \dots, v_{n-1} \}$ and $E(G) = \{ e_{ij} = \{ v_i, v_j \} \mid v_i \in V, v_j \in V \}$. Given parameter values are represented as a function $G(f) = \{ (v_i, f(v_i)) \mid v_i \in V, f(v_i) \in R \}$.

In search for good designs for topologically open dynamic systems, no topology can be evaluated in the absence of an associated set of parameters; and conversely, no set of parameter values can be evaluated except within the context of a given topology.

An obvious approach is to allocate to each new topology whatever amount of parameter search effort is needed to find at least a locally optimal parameterization, before judging the quality of the topology. A second strategy might be to allocate a fixed amount of search effort to each topology, adequate to optimize the parameters in many cases. However, such strategies may consume far more search effort than is practical during a simultaneous topology/parameter search, and when to stop each parameter search is difficult to determine.

2.2 Bond Graph Synthesis

Bond graph modeling is a powerful method that enables a unified approach to the analysis, synthesis and evaluation of dynamic system. It represents the common energy processes of multi-domain systems - electrical, mechanical, fluid, and thermal systems - in one graphical notation [5,6].

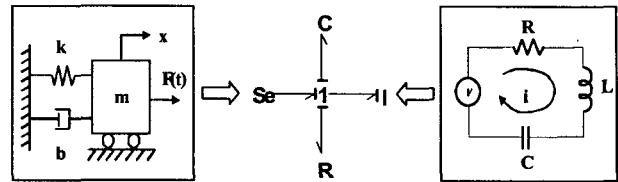


Fig. 1. The same bond graph model for two different domains

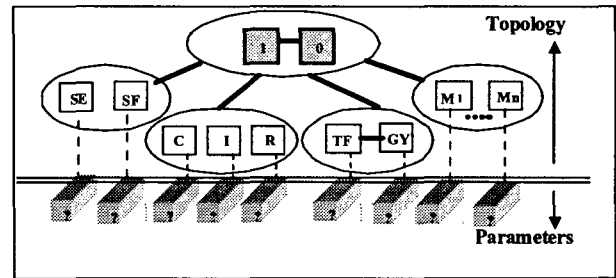


Fig. 2. The scheme of topology/parameter space in bond graphs

The scheme of topology/parameter search in bond graphs is as follows. Figure 2 shows component types typically used in bond graphs, including junctions, sources of effort or flow, one-port elements like resistors, capacitors and inductors, 2-port elements like transformers and gyrators, and higher-level modules composed of lower-level primitive elements. "Below the line" are associated parameters.

3. Hierarchical breeding control method

In this paper, we adopt a hierarchical breeding control mechanism to obtain better performance based on differential balancing of topology-altering operations and parameter-altering operations according to fitness level, in a fitness-structured multi-population model. The basic idea for this control mechanism arises from observing the human design process. Usually,

preliminary or conceptual design involves more structural modification, and final or detailed design involves more parameter tuning - i.e., there is greater concentration on design topology in the early stage and more on parameter tuning in the later stage.

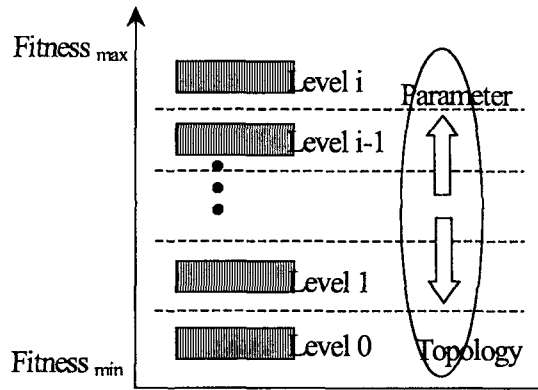


Fig. 3. Hierarchical breeding control structure. Subpopulations are organized in a hierarchy with ascending fitness levels.

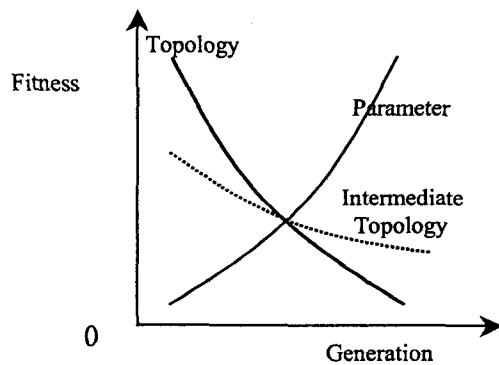


Fig. 4. Distribution of breeding control probability

4. Experiments and Analysis

To evaluate and compare the proposed approach with the previous one, the eigenvalue assignment problem is used, for which the design objective is to find bond graph models with minimal distance errors from a target set of eigenvalues. It is a classical "inverse" design problem - seeking something with given behavior, rather than the "forward" analysis problem of calculating the behavior of a given artifact.

4.1 Problem Definition

The following sets (consisting of various

6-, and 10-eigenvalue target sets, respectively) were used for the genetic programming runs:

Eigenvalue sets used in experiments:

- 1) $\{-12j, -2j, -30.5j\}$
- 2) $\{-10j, -110j, -33j\}$
- 3) $\{-20j, -120j, -77j\}$
- 4) $\{-1, -2, -3, -4, -5, -6\}$
- 5) $\{-20j, -120j, -77j, -124j, -412j\}$
- 6) $\{-1, -2, -3, -4, -5, -6, -7, -8, -9, -10\}$

The fitness function is defined as follows: pair each target eigenvalue one:one with the closest one in the solution; calculate the sum of distance errors between each target eigenvalue and the solution's corresponding eigenvalue, divide by the order, and perform hyperbolic scaling(not required with tournament selection, but making the numerical answers easier to interpret). Relative distance error (normed by the distance of the target from the origin) is used.

We used a strongly-typed version of lilgp [7] to generate bond graph models. These examples were run on a single Pentium IV 2.8GHz PC with 512MB RAM. The GP parameters were as shown below.

- Number of generations : 500
- Population sizes : 100 in each of ten subpopulations
- Initial population: half_and_half
- Initial depth : 3-6
- Max depth : 12 (with 800 max_nodes)
- Selection : Tournament (size=7)
- Crossover : 0.9
- Mutation : 0.1

The results of 6- and 10-eigenvalue runs are provided in Figures 5 and 6, showing average distance error for each set across 10 experiments. Figure 5 illustrates the comparison between the basic approach (without topology/parameter control) and the hierarchical topology/parameter breeding control on typical complex conjugate and real six-eigenvalue target sets. In all four sets, numbered 1)-4), the average error in the hierarchical topology/parameter breeding control approach is smaller than that of the basic approach. Figure 6 represents the results on two 10-eigenvalue sets, set numbers 5) and 6) above, and shows that the

new approach also outperforms the basic approach on these problems.

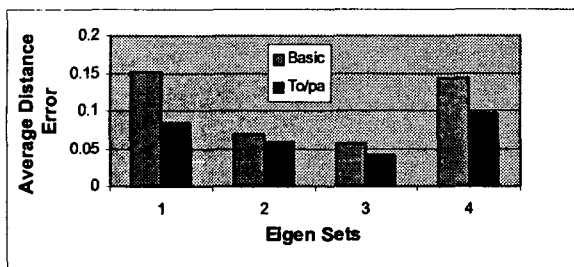


Fig. 5. Results for 6 eigenvalues

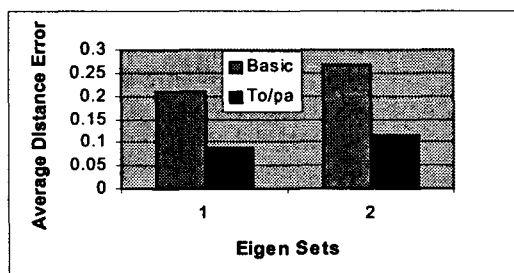


Fig. 6. Results for 10 eigenvalues (set 1 is eigenvalue set 5 above, etc.)

There are other factors to be determined to obtain optimal results, such as distribution of breeding control rate for topology- and parameter-altering operations, the control rate for the intermediate topology, and the ratio between fitness and generation etc. Current results are due to simple setting of control factors based on a few preliminary experiments. Therefore, much improvement is expected if optimal values of these control factors are found through further experiments and analysis.

5. Conclusion

This paper has introduced a hierarchical breeding control method for efficient topology/parameter evolution in bond-graph-based GP design. We adopt a hierarchical breeding control mechanism, implemented in a set of subpopulations separated hierarchically according to fitness levels, to obtain better performance based on balancing of topology/parameter search using a given set of switched modular primitives. Topology-altering operations are given higher

probability in high-fitness subpopulations, and parameter-altering operations get higher probability in lower-fitness subpopulations. Simultaneously, in all subpopulations, the percentage of topology-altering operations is reduced as the number of generations increases.

As a proof of concept for this approach, the eigenvalue assignment problem, which is to synthesize bond graph models with minimum distance errors from pre-specified target sets of eigenvalues, was used. Results showed better performance for all tested eigenvalue sets when the new topology/parameter control method was used. This tends to support the conjecture that a carefully tailored representation and sophisticated topology/parameter control method will improve the efficiency of GP search.

6. References

- [1] Koza, J. R. et al., Genetic Programming IV: Routine Human-Competitive Machine Intelligence. Kluwer Academic Publishers, 2003
- [2] Koza, J. R., Bennett F. H., Andre D., Keane M. A., Genetic Programming III, Darwinian Invention and Problem Solving. Morgan Kaufmann Publishers, 1999
- [3] Seo K., Hu, J., Fan, Z., Goodman E. D., Rosenberg R. C.: Automated Design Approaches for Multi-Domain Dynamic Systems Using Bond Graphs and Genetic Programming. Int. Jour. of Computers, Systems and Signals, 3(1). 2002 pp. 55-70
- [4] Tay E. Flowers W. and Barrus J., "Automated Generation and Analysis of Dynamic System Designs", Research in Engineering Design, vol 10, 1998, pp. 15-29.
- [5] Karnopp, D. C., Rosenberg R. C., Margolis, D. L., System Dynamics, A Unified Approach, 3rd ed., John Wiley & Sons (2000)
- [6] Sharpe J. E. E. E. , Bracewell R. H. , "The Use of Bond Graph Reasoning for the Design of Interdisciplinary Schemes", 1995 International Conference on Bond Graph Modeling and Simulation, 1995, pp. 116-121
- [7] Zongker, D., Punch, W., lil-gp 1.1 User's Manual. Michigan State University, (1996)