

Genetic Algorithm Design Inspired by Organizational Theory: Pilot Study of a Dependency Structure Matrix Driven Genetic Algorithm

Tian-Li Yu, David E. Goldberg, Ali Yassine, and Ying-Ping Chen

University of Illinois at Urbana-Champaign, Urbana, IL 61801
 {tianliyu,deg,yassine,ychen21}@uiuc.edu

Abstract. This study proposes a dependency structure matrix driven genetic algorithm (DSMDGA) which utilizes the dependency structure matrix (DSM) clustering to extract building block (BB) information and use the information to accomplish BB-wise crossover. Three cases: tight, loose, and random linkage, are tested on both a DSMDGA and a simple genetic algorithm (SGA). Experiments showed that the DSMDGA is able to correctly identify BBs and outperforms a SGA.

MDL-Based DSM Clustering. A dependency structure matrix (DSM) is a matrix where each entry d_{ij} represents the dependency between node i and node j . This study concentrates on the 0-1 domain, where $d_{ij} = 0$ means that there is no dependency between node i and node j , and 1 means that node i and node j are dependent. The goal of DSM clustering is to find subsets of DSM elements (i.e., clusters) so that nodes within a cluster are maximally dependent and clusters are minimally interacting.

The minimum description length principle (MDL) provides as a metric to cluster DSMs. Our MDL clustering metric is defined as follows:

$$f_{DSM}(M) = n_c \log(n_c) + \log(n_n) \sum_{i=1}^{n_c} cl_i + |S|(2 \log(n_n) + 1),$$

where n_c is the number of clusters, n_n is the number of nodes, and cl_i is the number of nodes in the i -th cluster. The objective is to find a model M that minimizes f_{DSM} .



Fig. 1. A slightly complicated DSM. Clustering is not so obviously at first glance.

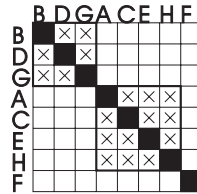


Fig. 2. After being reordered, the same DSM can be cleanly clustered as ((B,D,G)(A,C,E,H),(F)).

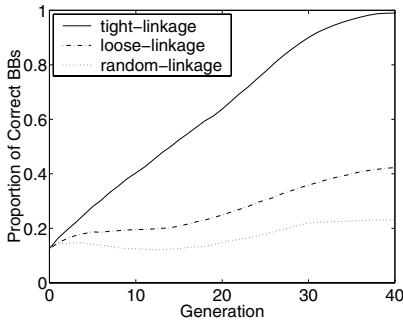


Fig. 3. The performance of the SGA with two-point crossover.

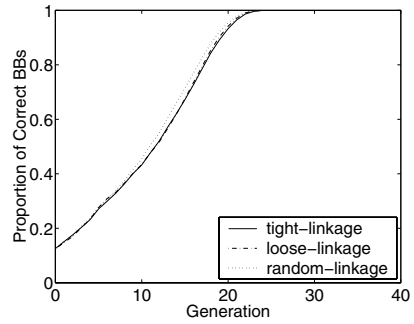


Fig. 4. The performance of the DSMDGA with BB crossover.

The DSMDGA. There are two levels of evolutionary algorithms in the DSMDGA: one is to solve the given problem, called the *primary GA*, and other one is to solve the building block (BB) identification problem, called the *auxiliary evolutionary strategy* (ES). The basic idea of the DSMDGA is to use the auxiliary ES to identify BBs, then the primary GA solves the problem by utilizing the BB information.

Empirical Results. The test function is a 30-bit Maxtrap problem composed of 10, 3-bit trap functions. Three linkage cases were tested: tight linkage, loose linkage, and random linkage. Figure 3 shows the performance of a simple GA (SGA) using two-point crossover. The SGA worked only for the tight linkage case. For loose and random linkage cases, SGA did not work because of BB disruption. Correspondingly, figure 4 illustrates the performance of the DSMDGA using BB-wise two-point crossover. The DSMDGA converged for all three tests. Even in the tight linkage test, the DSMDGA (converged at the 22th generation) outperformed the SGA (converged at the 40th generation) because the DSMDGA disrupted fewer BBs.

For more details of this study, please refer to Yu, Goldberg, Yassine, and Chen (2003).

Acknowledgment. This work was sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant F49620-00-0163, the National Science Foundation under grant DMI-9908252. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation thereon.

Reference

Yu, T.-L., Goldberg, D.E., Yassine, A., & Chen, Y.-P., (2003), *A genetic algorithm design inspired by organization theory: a pilot study of a dependency structure matrix driven genetic algorithm*. (IlliGAL Technical Report No. 2003007). Urbana, IL: University of Illinois at Urbana-Champaign.