ELSEVIER

Expert Systems with Applications 29 (2005) 705–712

Expert Systems with Applications

www.elsevier.com/locate/eswa

Two-phase sub population genetic algorithm for parallel machine-scheduling problem

Pei-Chann Chang*, Shih-Hsin Chen, Kun-Lin Lin

Department of Industrial Engineering and Management, Yuan-Ze University, 135 Yuan-Tung Road, Taoyuan 32026, Taiwan, ROC

Abstract

This paper introduces a two-phase sub population genetic algorithm to solve the parallel machine-scheduling problem. In the first phase, the population will be decomposed into many sub-populations and each sub-population is designed for a scalar multi-objective. Sub-population is a new approach for solving multi-objective problems by fixing each sub-population for a pre-determined criterion. In the second phase, non-dominant solutions will be combined after the first phase and all sub-population will be unified as one big population. Not only the algorithm merges sub-populations but the external memory of Pareto solution is also merged and updated. Then, one unified population with each chromosome search for a specific weighted objective during the next evolution process. The two phase sub-population genetic algorithm is applied to solve the parallel machine-scheduling problems in testing of the efficiency and efficacy. Experimental results are reported and the superiority of this approach is discussed.

Keywords: Scheduling problem; Genetic algorithm; Multi-objective optimization; Evolution strategy

1. Introduction

The literature of parallel machine-scheduling problems has been extensively reviewed by Cheng and Sin (1990). Garey and Johnson (1979) showed that scheduling jobs on two identical machines to minimize the makespan is NP-hard (NP is the abbreviation of Non-Polynomial). As identified by Brucker (1998) when the number of machine is greater than two, the problem is even strong NP-hard. Therefore, the parallel machine-scheduling problem presents a great challenge to the industrial practitioners and academic researchers. As a result, efficient heuristic algorithm should be developed in order to deal with practical industrial scheduling problems especially in drilling operation-scheduling problems of Printed Circuit Board (PCB) industries as presented by Hsieh, Chang, and Hsu (2003).

Owing to the development in genetic algorithm, it provides a new method and new direction for scheduling researchers to apply this new tool. Successful application examples can be found in Carlos and Peter (1995), Chang, Hsieh, and Lin (2002), Chang, Hsieh, and Wang (2003), Lo and Bavarian (1992), Neppali, Chen, and Gupta (1996), Sridhar and Rajendran (1996), and Wang (2003, 2005). However, in practical application the goal in PCB industries is always multi-objective which includes makespan, due-date and flowtime. Recent development in Evolutionary Multi-objective Optimization provides interesting results as discussed by Deb, Amrit Pratap, and Meyarivan (2000) and Zitzler, Laumanns, and Bleuler (2004). In that, different EMO algorithms are proposed and efficiency and solution quality are greatly improved.

Inspired by these pioneer works as discussed above, this research proposes a two phase sub-population genetic algorithm to solve the parallel machine-scheduling problem. In the first phase, the population will be decomposed into many sub-populations and each sub-population is designed for a scalar multi-objective. Sub-population is a new approach for solving multi-objective problems by fixing each sub-population for pre-determined criteria. In the second phase, non-dominant solutions will be combined after the final evolutions. One unified population by combining these sub-populations will be applied for regular evolution.

The rest of the paper is organized as follows: Section 2 gives literature review. Section 3 introduces the TPSPGA

* Corresponding author. Tel.: +886 3 4352654; fax: +886 3 4559378.
E-mail address: iepchang@saturn.yzu.edu.tw (P.-C. Chang).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2005.04.033

algorithm. Then experimental results are given in Section 4. Finally, the conclusion is discussed and performance of the algorithm is evaluated.

2. Literature review

The genetic algorithm has been widely discussed. The following related works present the efforts of multi-objective genetic algorithm and its application on scheduling problem.

Holland (1975) proposed the Genetic Algorithm (GA) that imitates the natural evolution progress, including the selection, crossover, and mutation. The procedure produces better offspring in accommodating to the environment. Schaffer (1985) proposed VEGA (Vector Evaluated Genetic Algorithm) to solve the Pareto-optimal solution of multi-objective problem. The VEGA is the first method modifying the GA to solve multi-objective problems. It selects better chromosomes from separate sub-mating pools so that the selected chromosomes satisfy different objectives. However, the drawbacks of the algorithm are: (1) the algorithm ca not guarantee that all the solutions are Pareto-optimal solution; and (2) the algorithm cannot keep the diversity of the solutions during the evolving process.

Muruta and Ishibuchi (1996) employed the structure of genetic algorithm in searching the multi-objective problem, and the algorithm is named MOGA (Multi-Objective Genetic Algorithm). One characteristic of MOGA is using the dynamic weighting to transform the multiple objectives into single objective, which randomly assigns different weight value to different objectives. The algorithm also applies the elite preserving strategy that randomly selects chromosomes from Pareto set. This technique prevents from sinking into local optimal. Muruta, Ishibuchi, and Tanaka (1996) found the MOGA was superior to VEGA or GA on multi-objective problem of flow shop scheduling problem. NSGA2 (Non-dominated Sorting Genetic Algorithm-II) was proposed by Deb et al. (2000), where the Elitism strategy was adopted. Besides, in order to keep the solution diversity, the algorithm also provided a crowding distance to measure the density of individuals in solution space.

Coello Coello et al. (2001) proposed micro-GA, which refers to small-population genetic algorithm with reinitialization. They found that micro-GA is able to converge to better solution although there are few individuals in each population. The more individuals in a population, algorithm needs more computational time. Therefore, it saved the computational effort and increases the efficiency. The performance of GA that accompanied vector evaluated approach (from the concept of Schaffer, 1985) and weighted criteria approach (linear combination of objectives) on multi-objective scheduling problem was evaluated by Neppali et al. (1996). The evaluation result showed that the presented vector evaluated approach was better than weighted criteria approach. Muruta and Lshibuchi (1994) utilized MOGA to solve the flowline scheduling problem, which emphasizes on random weight assignment and Elitism. The research pointed out that the Elitism was able to find out the near Pareto set more efficiently and fast.

Funda and Ulusoy (1999) employed the GA and considered the total weighted earliness and tardiness on the mutliobjective scheduling problem of parallel machine. Their suggestion is that if the local search was included in the GA, the performance may become better. Gupta, Neppalli, and Werner (2001) discussed the multi-objective scheduling problem of parallel machines, which attempts to minimize the make span under the consideration of minimizing the flow time. They proposed Two-Machine Optimization Procedure, Longest Processing Time Procedure, Multi-fit Procedure, and Hierarchical Criteria Algorithm. The experiment result showed the combination of Hierarchical Criteria Algorithm and Multi-fit Procedure was more efficient when compared to others combinations.

Motivated by the literature discussed above, this research introduces a two-phase sub population genetic algorithm to solve the parallel machine-scheduling problem. Detailed procedures of TPSPGA are presented in Section 3.

3. Two phase sub-population genetic algorithm (TPSPGA)

According to Simoes and Costa (2002), the loss of diversity may mean pre-mature of evolution algorithm. In order to prevent the searching procedure from being trapped into local optimality, this research proposes a two phase sub-population genetic algorithm (TPSPGA) to solve the parallel machine-scheduling problems. The basic idea of TPSPGA is to decompose the population into several sub-populations, which are assigned different weights by scalarizing multiple objectives into single objective. Each sub-population is just like a squad team with a pre-assigned goal and hopefully they will be marching in the right direction to reach the high peak of the landscape in terms of fitness performance. By the effort, every sub-population concentrates on specific exploring space and thus the diversity of the population can be kept among these sub-populations. After certain evolutions of the searching process, to further improve the solution quality, all the sub-population will be reunified and each individual chromosome will be reassigned a scalarized objective to further expand the searching process until the final near-optimal solution is found.

As discussed above, there are three main characteristics of the TPSPGA method: (1) numerous small sub-populations are designed to explore the solution space; (2) the multiple objectives are scalarized into single objective for each sub-population; and (3) two phase implementation is applied to deal with the gradually narrowing down of the searching space. The framework of TPSPGA is illustrated in