Knowledge-Based Systems 57 (2014) 95–103

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A novel fruit fly optimization algorithm for the semiconductor final testing scheduling problem

Xiao-long Zheng, Ling Wang *, Sheng-yao Wang

Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Automation, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 16 September 2013
Received in revised form 3 December 2013
Accepted 9 December 2013
Available online 16 December 2013

Keywords:
Fruit fly optimization algorithm
Smell-based search
Vision-based search
Cooperative search
Semiconductor final testing scheduling problem

ABSTRACT

In this paper, a novel fruit fly optimization algorithm (nFOA) is proposed to solve the semiconductor final testing scheduling problem (SFTSP). First, a new encoding scheme is presented to represent solutions reasonably, and a new decoding scheme is presented to map solutions to feasible schedules. Second, it uses multiple fruit fly groups during the evolution process to enhance the parallel search ability of the FOA. According to the characteristics of the SFTSP, a smell-based search operator and a vision-based search operator are well designed for the groups to stress exploitation. Third, to simulate the information communication behavior among fruit flies, a cooperative search process is developed to stress exploration. The cooperative search process includes a modified improved precedence operation crossover (IPOX) and a modified multipoint preservative crossover (MPX) based on two popular structures of the flexible job shop scheduling. Moreover, the influence of the parameter setting is investigated by using Taguchi method of design-of-experiment (DOE), and suitable values are determined for key parameters. Finally, computational tests results with some benchmark instances and the comparisons to some existing algorithms are provided, which demonstrate the effectiveness and the efficiency of the nFOA in solving the SFTSP.

1. Introduction

As one of the most important high-tech industries, nowadays the semiconductor manufacturing is confronted with many opportunities and challenges simultaneously. The rapid growth of the semiconductor industry brings greater economic benefits. Meanwhile, the fierce competition urges the semiconductor company to increase the productivity and utilize resources effectively. As the final operation in semiconductor manufacturing [1], final testing of integrated circuit (IC) products is used to test whether the products meet the requirements of the standard specifications [2]. Due to the demand fluctuation and the high cost, usually testing machines are very limited. The semiconductor final testing scheduling problem (SFTSP) [1] is to determine a feasible schedule with the satisfactory makespan while utilizing the limited testing resources efficiently. As a specific type of simultaneous multiple resources scheduling problem, the SFTSP is one of the common NP-hard scheduling problems [1,3]. Currently, most semiconductor companies only use empirical methods composed of several heuristic rules to conduct final test scheduling. As a result, the obtained solutions are often local optima [2]. Due to the significance of the SFTSP in both academic research and real applications, it is important to develop novel intelligent and knowledge-based algorithms for achieving satisfactory performances.

During the past two decades, some research work has already been done for solving the SFTSP. Relevant results in this area include the following. In [4], a suitable scheduling system was developed, which characterized the operations in the semiconductor testing facility by a broad product mix, variable lot sizes and yields, long and variable setup times, and limited test equipment capacity. In [5], a branch and bound method with Lagrangian Relaxation (LR) was employed to solve the SFTSP. The feasibility of the LR was tested with two randomly generated examples and two real factory cases. In [6], decomposition methods were presented for the SFTSP by exploiting the structure of the routings in semiconductor testing to decompose the shop into a number of work centers, and some specialized procedures were used to schedule the centers. In [7], two Petri net-based hybrid heuristic strategies were presented by combining the heuristic best-first strategy with the controlled backtracking strategy. In [8], the multi-head tester scheduling problem was described and an enumeration procedure was designed by considering different number of heads. In addition, the SFTSP was modeled as an unrelated parallel machine problem with multiple constraints in [9], and theory of constraints was employed to develop a heuristic capacity-constrained scheduling algorithm.

Recently, several meta-heuristic algorithms were used to solve the SFTSP. In [1], a mathematical mixed-integer linear programming model (MILP) was constructed for the SFTSP to optimize

* Corresponding author. Tel.: +86 10 62783125; fax: +86 10 62786911.
E-mail address: wangling@mail.tsinghua.edu.cn (L. Wang).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.12.011

the testing job scheduling, and a genetic algorithm (GA) was proposed to solve the problem in a short time for practical viability. In [10], a novel bi-vector encoding method based GA (bvGA) was developed, where the operation sequence and the seizing rules for resource assignment in tandem are both represented. In [3], a cooperative estimation of distribution algorithm (CEDA) was developed for the SFTSP by extending a co-evolutionary framework with a divide-and-conquer strategy. It showed that the CEDA could be more effective than both the GA [1] and the bvGA [10].

As a relatively new evolutionary optimization approach, fruit fly optimization algorithm (FOA) [11] is inspired by the knowledge from the foraging behavior of real fruit flies. The FOA has few parameters to adjust, and it is easy to understand and implement. Due to its merits, the FOA has already been successfully applied to solve some academic and engineering optimization problems, including financial distress [11], web auction logistics service [12], general regression neural network optimization [13], and proportional integral derivative controller tuning [14,15]. However, to the best of our knowledge, the FOA has not been applied to any scheduling problem yet. The SFTSP is an NP-hard combinatorial optimization problem with a huge search space, which is difficult to be solved effectively by traditional methods. The FOA is an evolutionary algorithm with a parallel search framework by using a smell-based search and a vision-based search. And problem-dependent local search can also be easily incorporated into the search framework of the FOA to further enhance exploitation. Besides, a cooperative search can be designed by sharing the information among fruit flies to stress exploration. So, it could be powerful to solve the SFTSP by reasonably designing the FOA. With such a motivation, we will propose a novel FOA (nFOA) in this paper to solve the SFTSP. To be specific, a new encoding scheme is presented to represent solutions reasonably, and a new decoding scheme is presented to generate feasible schedules. Based on the characteristics of the problem, a smell-based search and a vision-based search are designed for multiple fruit fly groups to perform evolutionary search. According to the similar characteristics between the SFTSP and the flexible job shop scheduling (FJSP) [16–18], the improved precedence operation crossover (IPOX) and the multipoint preservative crossover (MPX) are adopted and modified for a cooperative search to simulate the information communication behavior among fruit flies. To investigate the influence of parameter setting, Taguchi method of DOE is carried out. Finally, the effectiveness of the proposed nFOA is demonstrated by computational tests with benchmark instances and comparisons to some existing algorithms.

The remainder of the paper is organized as follows: In Section 2, the SFTSP is described. In Section 3, the nFOA for solving the SFTSP is presented in details after introducing the basic FOA briefly, and the computational complexity is analyzed. In Section 4, the influence of parameter setting is investigated based on DOE testing, and the computational results and comparisons are provided. Finally, the paper is ended with some conclusions and future work in Section 5.

2. Problem description

The SFTSP can be described as follows. There are n jobs (IC products) $J = \{J_1, J_2, \ldots, J_n\}$ to be tested on m machines $M = \{M_1, M_2, \ldots, M_m\}$. Usually, different jobs may require different steps of functional testing due to different product specifications. Each job J_i consists of a sequence of n_i operations $\{O_{i,1}, O_{i,2}, \ldots, O_{i,n_i}\}$, such as functional test, burn-in, scan, bake, tape and reel, and package and load. Each operation $O_{i,j}$ ($i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n_i$) can be executed on one machine out of its capable machine set $M_{i,j} \subseteq M$. The processing time of an operation on different execution machine may be different. In addition, usually it assumes that: (1) All jobs and machines are available at time 0; (2) preemption is not allowed, i.e., each operation must be completed without interruption once it starts; (3) there is no precedence constraint among the operations of different jobs; (4) at one time a machine can execute at most one operation, and it is available to other operations only if the previous operation is completed.

Compared to the well-known flexible job shop scheduling [16–18], the SFTSP has the following additional characteristics.

(1) Simultaneous multi-resources. In addition to the machines, extra resources are needed, i.e., the tester, handler and accessory [1,3,10]. A specific operation of the job can only be carried out with appropriate configuration of resources [3]. Moreover, these resources are prepared in limited amounts due to the high price [3,19].
(2) Sequence-dependent setup time. It is inevitable to cost time to complete setup activities comprising resources assembly and disassembly, temperature change, software download, and calibration in real semiconductor testing. Thus, a sequence-dependent setup time is further required to disassemble the original machine and to assemble and calibrate the new machine for the incoming operation [1,3]. In addition, the setup is separable from process and anticipatory so that a setup can start before the job is ready [10].

Essentially, the SFTSP can be regarded as a simultaneous multi-resources flexible job-shop scheduling problem with sequence-dependent setup time, as illustrated in Fig. 1. The objective of the SFTSP is to determine both the assignment of machines and the sequence of operations on all the machines to minimize the maximum completion time of all the operations (i.e., makespan), without violating the resources constraints at any time.

3. nFOA for SFTSP

In this section, the basic FOA will be introduced first. Then a novel FOA (nFOA) will be presented for solving the SFTSP, including the encoding scheme, decoding method, population initialization, smell-based search, vision-based search, and cooperative search process. Finally, the procedure of the nFOA will be provided and its computational complexity will be analyzed.

3.1. Basic FOA

The FOA [11] is an interactive evolutionary method with two particular idealized search processes based on the foraging characteristics of the fruit flies: locate the food source through smelling and fly towards the corresponding location (i.e., smell-based search process), and fly towards the food source up close using sensitive vision (i.e., vision-based search process). The foraging process of a fruit fly group can be illustrated in Fig. 2, and the procedure of the FOA is summarized as follows [11].

Step 1. Parameters initialization: set the maximum number of generations and population size.
Step 2. Fruit fly group initialization.
Step 3. Smell-based search process: randomly generate several fruit flies around the fruit fly group to construct a population.
Step 4. Evaluation: evaluate the population to obtain the smell concentration values (fitness value) of each fruit fly.
Step 5. Vision-based search process: find the best fruit fly with the maximum smell concentration value, and then let the fruit fly group fly towards the best one.