Expert Systems with Applications 38 (2011) 5438–5447

Contents lists available at ScienceDirect

Expert Systems with Applications

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

A new bee colony optimization algorithm with idle-time-based filtering scheme for open shop-scheduling problems

Yueh-Min Huang [a,b,*], Jin-Chen Lin [a]

[a] Department of Engineering Science, National Cheng Kung University, No. 1, Ta-Hsueh Road, Tainan 701, Taiwan, ROC
[b] Department of Applied Geoinformatics, Chia Nan University of Pharmacy and Science, 60, Erh-Jen Rd., Sec. 1, Pao-An, Jen-Te, Tainan 717, Taiwan, ROC

ARTICLE INFO

Keywords:
Bee colony optimization
Open shop scheduling

ABSTRACT

Open shop scheduling problems (OSSP) are one of the most time-consuming works in scheduling problems. Currently, many artificial intelligence algorithms can reduce the problem-solving time to an acceptable time range, and even can further downsize the range of solution space. Although the range of solution space is technically downsized, in most scheduling algorithms every partial solution still needs to be completely solved before this solution can be evaluated. For example, if there is a schedule with 100 operations, then all 100 operations must be scheduled before the scheduler can evaluate its fitness. Therefore, the time–cost of unnecessary partial solutions is no longer saved.

In order to improve the weakness stated above, this paper proposes a new bee colony optimization algorithm, with an idle-time-based filtering scheme, according to the inference of "the smaller the idle-time, the smaller the partial solution", and "the smaller the makespan (Cmax) will be". It can automatically stop searching a partial solution with insufficient profitability, while the scheduler is creating a new scheduling solution, and therefore, save time–cost for the remaining partial solution. The architecture and details of the bee colony optimization heuristic rule is detailed in this paper.

1. Introduction

The shop-scheduling problem can be simply introduced as a problem of redistribution of resources or a problem of rearrangement of operations order. Open shop scheduling problem (OSSP) is more difficult than JSSP and FSSP because its operations have no predefined order. When $m >= 3$, OSSP is proved as a NP-complete problem (Garey & Johnson, 1979; Gonzalez & Sahni, 1976).

In recent years, many heuristic rules for solving OSSP have been presented. One of the earliest heuristic rules is the branch and bound method, which regards the solution space as a tree with limited branches. Through the bound design of scientists, such as lower bound (the lower bound of a solution), the searching range of the solution space can be effectively downsized. The advantage of the branch and bound method is that the solution space is a clear tree structure, whose branches can be examined without omission to find a feasible solution. Its weakness is that the changes of feasible solutions are concentrated on certain branches, thus, that its feasible solution has an inherent lack of changeability.

Intelligent heuristic rules for various colonies have aroused widespread exploration and application in recent years, such as GA, ACO, PSO, BCO, and their combinations. These heuristic rules integrate the techniques of random a number generator, parallel operations, probability rules, fitness functions, tabu serial, etc., and in the face of a problem with large solution space, attempt to solve problems, where a feasible solution lacks changeability, and to effectively exclude bad solutions more quickly, in order to come close to the optima or even obtain the optima.

Louis and Xu (1996) proposed to integrate the concepts of GA and CBRP, of which CBRP is directed at special cases to provide specific problem-solving inference principles. Prins (2000) suggested that, the characteristics of individuals in a GA colony were differed from each other, and their chromosomes could be rearranged to be applicable to the detection of a global optima from quasi-optimal schedules, meaning a feasible solution with diversity, found through GA, could meet the needs of feasible solutions from a large solution space. Liaw (2000) developed a HGA that incorporated a local improvement procedure, based on TS, for solving the open shop scheduling problem. The incorporation of the local improvement procedure enabled the algorithm to perform a genetic search over the subspace of the local optima, and then, TS performed the local improvement procedure, which costs heavy computing time on a computer, and therefore, it is only performed in several rounds of the local improvement procedure. Pan and Huang

* Corresponding author at: Department of Engineering Science, National Cheng Kung University, No. 1, Ta-Hsueh Road, Tainan 701, Taiwan, ROC. Tel.: +886 6 2757575x63336; fax: +886 6 2388123.

E-mail address: huang@mail.ncku.edu.tw (Y.-M. Huang).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.10.010

(2009) proposed a hybrid genetic algorithm (HGA) for no-wait job shop scheduling problems. In this paper, the chromosome was represented by a vector of integer numbers. The hybridizing method was the Order Crossover adopted by Brizuela et al. The mutation method was to randomly exchange two genes in the same chromosome. A genetic operation was defined by cutting out a section of genes from a chromosome, for treatment as a sub problem. This sub problem was then transformed into an asymmetric traveling salesman problem (ATSP), and solved with a heuristic algorithm. Subsequently, this section, with a new sequence, was returned to replace the original section of chromosome. The experimental results showed that this hybrid genetic algorithm could accelerate the convergence and improve solution quality.

Blum (2005) hybridized ant colony optimization (ACO) with a beam search to overcome difficult combinatorial optimization problems. This method caused the probability ACO mechanism to produce a group of complete solutions, and then used the beam search to perform improvement procedures of partial solutions. This method could randomly search the solution space and direct the solution to optimal branches.

Sha and Hsu (2008) modified the particle position representation using priorities, and the particle movement using an insert operator, and implemented a modified parameterized active schedule generation algorithm (mP-ASG) to decode a particle position in a schedule. In mP-ASG, the search area between non-delay schedules and active schedules could be reduced or increased by controlling the maximum delay time allowed. Zhang and Sun (2009) proposed an alternate two-phase particle swarm optimization algorithm, called ATPPSO, to solve flow shop-scheduling problems, with an objective of minimizing makespan, which included two processes, the attraction process, and the repulsive process, which executed alternatively. In the attraction process, every particle was attracted to the individual optimum location and current global optimum location. In the repulsive process, every particle could avoid the worst individual location, which could not only diversify the colonies, but also increase the speed to the convergence of ATPPSO algorithm. Kuo et al. (2009) proposed an efficient a new hybrid particle swarm optimization model, named HPSO, to solve flow shop-scheduling problems. HPSO combined a random-key (RK) encoding scheme and an individual enhancement (IE) scheme. The RK encoding scheme could encode the locations in RK virtual space as locations in a FSSP solution space. In the RK virtual space, each location was represented by a vector of real numbers, while in FSSP solution space, each location was represented by a vector of integer numbers. For example, locations (0.3, 0.5, 0.2, and 0.4) in the RK virtual space could be encoded as locations (3, 1, 4, and 2) in the FSSP solution space. The advantage of using the RK encoding scheme was that it could make full use of the query ability of PSO. The idea of an IE scheme was to exchange the order of two operations in the same job, and if the makespan after the exchange was good, it was retained. Using an IE scheme could enhance the local search ability of particle swarm.

Chong et al. (2006) proposed a novel approach that used the honeybee foraging model to solve job shop-scheduling problems. This approach decided whether the branch from node A or node B was formed by the side length ratio between nodes A and B, as well as the heuristic distance between nodes A and B. The branching probability was calculated according the average profitability of the previous round. Wong et al. (2008) published the Big Valley Landscape Exploitation (BVLE) method, which proposes to define a sole search space, called a landscape, when a heuristic search approach was applied to a combinatorial optimization problem. Exploring the search space with different numbers of search operations could create different landscapes, as the content of the landscape might change with the number of heuristic operations. The structures of these landscapes could help search for the global optima. In the BVLE structure, the local optima in some clusters may tend to appear near other local optima, and every cluster would form a valley centered on the global optimum. BVLE suggested that the new starting point for a search should be based on the previous local optimum, other than a random point in the search, because good candidate solutions can often be found near a local optimum.

This study proposes a new bee colony optimization algorithm, with an idle-time-based filtering scheme, which can automatically stop searching a partial solution with insufficient profitability, while the scheduler is creating a new scheduling solution, and therefore, save on time–cost for the remaining partial solution.

The remainder of this paper is organized as follows: Section 2 introduces OSSP; Section 3 describes the born foraging behaviors of a bee colony; Section 4 defines the computerized bee colony behaviors; Section 5 describes the origin and development of ASG, and then summarizes the mP-ASG used in experiments; Section 6 discusses the experimental results.

2. The open shop scheduling problem (OSSP)

Shop-scheduling problems (SSP) can be introduced simply as problems of redistribution of resources, or a problem of rearrangement of operational order. SSP is mainly divided into following three types: 1. FSSP, 2. JSSP, and 3. OSSP.

As far as the difficulty for solving, OSSP is more difficult than JSSP and FSSP, due to the fact that it has no predefined order of operations in the same job. When $m >= 3$, OSSP is proved to be an NP-complete problem (Garey & Johnson, 1979; Gonzalez & Sahni, 1976).

In this paper, the definitions (Brucker, Hurink, Jurisch, & Wostmann, 1997; Gueret & Prins, 1998; Liaw, 1999) of OSSP are summarized, as follows:

Suppose the existing resources are m sets of machines, and the jobs are j pieces.

(1) Each job contains m operations.
(2) Each operation must cost time pij.
(3) The operations in a same job are randomly ordered, but at a time, only an operation can be handled.
(4) No preemptive, which indicates that no operation can interrupt other operations.
(5) At one time, a machine can handle only one operation, and each job can only be handled by one machine.

In this paper, the problem is to minimize the makespan (Cmax) of OSSP. Here, makespan has the same meaning as Cmax, that is, the time needed for completing all jobs.

3. The native foraging behavior of a honeybee colony

Karl von (The 1973 Nobel Prize winner) and his students (Karl von, 1967) decoded the dance language of bees. They noted that the direction information of a bee dance indicated the location of a food source, relative to the sun, and the distance of the food source was signaled by different kinds of dance. Tarpy (2009) and Wenner and Wells (1990) argued that floral odors on a forager's body were the primary cues that enabled the recruit-bees to locate new food sources. Either dance languages or floral odors indicate that there is communication between bees that fulfill foraging behaviors.

The self-organization of bees (Camazine & Sneyd, 1991) is based on several simple behavioral rules of insects. The best examples are locating and collecting nectar. Each bee collects nectar according to the route indicated by the forager bee that found the foraging route. Each hive has a dance area where the forager bee that found