An Efficient Meta-heuristic based on Self-control Dominance Concept for a Bi-objective Re-entrant Scheduling Problem with Outsourcing

Atefeh Moghaddam*, Farouk Yalaoui, and Lionel Amodeo

Charles Delaunay Institute (ICD-LOSI), University of Technology of Troyes,
STMR, UMR CNRS 6279,
12 rue Marie Curie, 10010 Troyes, France
{atefeh.moghaddam,farouk.yalaoui,lionel.amodeo}@utt.fr

Abstract. We study a two-machine re-entrant flowshop scheduling problem in which the jobs have strict due dates. In order to be able to satisfy all customers and avoid any tardiness, scheduler decides which job shall be outsourced and find the best sequence for in-house jobs. Two objective functions are considered: minimizing total completion time for in-house jobs and minimizing outsource cost for others. Since the problem is NP-hard, an efficient genetic algorithm based on modified self-control dominance concept with adaptive generation size is proposed. Non-dominated solutions are compared with classical NSGA-II regarding different metrics. The results indicate the ability of our proposed algorithm to find a good approximation of the middle part of the Pareto front.

Keywords: Scheduling, re-entrant, bi-objective, outsourcing, genetic algorithm, dominance area

1 Introduction

In today's competitive market, one of the most important survival factor for a company is the achievement of customer satisfaction which guarantees its long-run financial performance. Due to the resource constraints and clients' requirements, manufacturers are not always able to meet customers' due dates so tardiness is occurred. Outsourcing is an alternative to avoid loosing clients. In this paper, we study a bi-objective two-machine re-entrant permutation flowshop scheduling problem in which completing an order after its due date is not allowable so that order will be outsourced. Recent literature surveys on re-entrant scheduling problems and outsourcing can be found in [1] and [2] respectively.

The system that we study in this paper is illustrated in Fig. 1. For each job, its processing time on both machines on both cycles, due date and outsourcing cost are known in advance and the processing route for in-house jobs is

* Corresponding author.

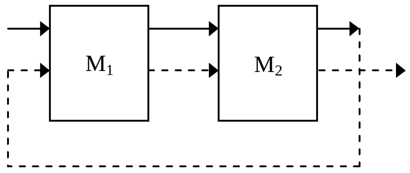

Fig. 1. Re-entrant flowshop scheme

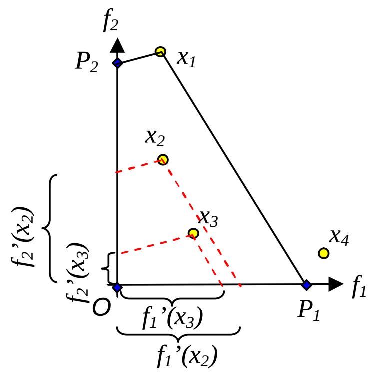

Fig. 2. Dominance area regarding m-SCD

(M_1, M_2, M_1, M_2). Other assumptions are the same as classical scheduling problem (i.e. no preemption, no break-down, ...). We are looking for a set of non-dominated solutions regarding two objective functions: minimizing total completion time for in-house jobs and minimizing outsourcing cost. Since the simpler variant of our problem is NP-hard [3], we propose a genetic-based algorithm inspired from NSGA-II [4] with different dominance concept than Pareto and adaptive generation size. The proposed dominance concept is the modification of Self-Control Dominance Area of Solutions (S-CDAS) introduced by Sato et al. [5].
We show that our algorithm is able to find limited number of non-dominated solutions compared to NSGA-II but with higher quality.

2 Modified Self Control Dominance Concept

Since many years ago, Pareto-dominance concept has been integrated to different meta-heuristics to find a good estimation of set of non-dominated solutions. Recently, it has been shown that other dominance properties rather than Pareto, may help algorithms to find better estimation of non-dominated solutions, e.g., Lorenz-dominance [6] and Self-Control Dominance Area of Solutions [5]. Self-Control Dominance Area of Solutions (S-CDAS) introduced for the first time by Sato et al. [5]. They showed that by integrating S-CDAS into NSGA-II, better estimation of non-dominated solutions for multi-objective 0-1 knapsack problem could be found compared to NSGA-II with Pareto dominance. In this paper we introduce a new dominance concept inspired from S-CDAS which is called modified SCD (m-SCD). In S-CDAS the objective is to find better estimation of *whole* pareto front so the parameters are set in a way to keep the extrema in each Pareto non-dominated front. On the contrary, in m-SCD we focus our search on the middle part of the Pareto front. The trade-off characteristic renders such area of particular interest in practical applications. So we try to find better estimation of non-dominated solutions located in this area.

In m-SCD, the concept is to make Pareto-non-dominated solutions different one from another by inducing more fine-grained ranking in order to be converged into the middle part of Pareto-front.

Definition: modified Self-Control Dominance (m-SCD)
In a bi-objective minimization problem, solution x dominates solution y based on m-SCD properties ($x \prec_{m-SCD} y$) if *one* of the following statements holds true:

- x dominates y in Pareto sense ($x \prec_P y$); or
- x and y are Pareto-equivalent and $SCD(x) \prec_P SCD(y)$ where $SCD(x) = (f'_1(x), f'_2(x))$ in which $f'_1(x)$ and $f'_2(x)$ are derived from non-orthogonal projection of point x onto xy-plane as described in more details in step 2-2.

In Fig. 2, although x_1, x_2, x_3, x_4 are all Pareto non-dominated solutions, x_1 and x_4 which are the extrema are dominated by x_2 and x_3 based on m-SCD concept. In the following, we describe step by step how we calculate the values of different parameters shown in Fig. 2 to reach the values of f'_1 and f'_2.

Step 1: Consider a set of Pareto non-dominated solutions $X = \{x_1, x_2, ...x_k\}$. We define the parameters as below:

$$O = (f_1^{min} - \varepsilon, f_2^{min} - \varepsilon) \tag{1}$$

$$P_1 = (f_1^{max} - \varepsilon, f_2(O)) \tag{2}$$

$$P_2 = (f_1(O), f_2^{max} - \varepsilon) \tag{3}$$

where f_i^{min} (f_i^{max}) is the minimum (maximum) value of the i-th objective function in the set X and ε is a tiny constant.

Step 2: For each solution $x_j \in X$ $(j = 1, 2, ..., k)$ we repeat the following steps:

Step 2-1: Find the slope of both lines through two pairs of points x_j, P_1 and x_j, P_2 (accordingly use SLOPE1 and SLOPE2 as the values). We have:

$$SLOPE1 = \frac{f_2(x_j) - f_2(P_1)}{f_1(x_j) - f_1(P_1)} \tag{4}$$

$$SLOPE2 = \frac{f_2(x_j) - f_2(P_2)}{f_1(x_j) - f_1(P_2)} \tag{5}$$

$$SCD(x_j) = (f'_1(x_j), f'_2(x_j)) = (f_1(P_1), f_2(P_2)) \tag{6}$$

In other words, x-value of P_1 and y-value of P_2 are the same as $f'_1(x_j)$ and $f'_2(x_j)$.

Step 2-2: For each solution $y \in X - \{x_j\}$, we calculate $f'_1(y)$ and $f'_2(y)$ by projecting point y onto the x-axis regarding SLOPE1 and onto the y-axis regarding SLOPE2.

$$f'_1(y) = f_1(y) + \frac{f_2(O) - f_2(y)}{SLOPE1} \tag{7}$$

$$f'_2(y) = f_2(y) + SLOPE2(f_1(O) - f_1(y)) \tag{8}$$

$$SCD(y) = (f'_1(y), f'_2(y)) \tag{9}$$

Step 2-3: Regarding the new values calculated for each member of set X, we use Pareto-dominance properties to find the solutions that dominate x_j.

3 Experimental Results

We conduct experiments on 7 randomly generated problems with 10, 15, 20, 40, 50, 70 and 100 jobs to test the performance of our proposed algorithm. Processing times are generated from the discrete uniform distribution within a range of [1,100] on both machines. Due dates of the jobs are generated using two parameters, T (tardiness factor) and R (due date range) as described in [7] with more details. In this paper we set T=0.3 and R=1.4. The rejection costs are calculated based on the function $exp(5 + \sqrt{a} \times b)$, where a is a random integer number within [1,80] and b is a random number within [0,1].
Our proposed algorithm is based on NSGA-II coupled with m-SCD dominance with adaptive generation size proposed by Tan et al. [8] in which chromosome repairing is done by eliminating the job with minimum outsourcing cost scheduled before first tardy job. This algorithm is compared to classical NSGA-II with Pareto dominance and fixed number of generation in which the first tardy job detected in schedule is eliminated for making the solution feasible. The results on 7 different instances show that on the average, 31% of solutions found by NSGA-II are dominated by m-SCD-NSGA-II while 4% of solutions found by m-SCD-NSGA-II are dominated by NSGA-II. The hypervolume ratio of NSGA-II to m-SCD-NSGA-II is 0.87 in average which implies that the area dominated by m-SCD-NSGA-II is larger than NSGA-II dominance area. In addition, the solutions found by m-SCD-NSGA-II are distributed more evenly than those found by NSGA-II. However, the spread of solutions in NSGA-II is significantly more than those solutions achieved by our proposed algorithm. This fact is completely in-line with our parameters definitions in m-SCD. Regarding the computational time, our proposed algorithm is in average 4 times slower than classical NSGA-II. The reason is first due to the additional computational effort for calculating dominance area based on m-SCD and also increasing in number of generation.

4 Conclusion

In this paper we studied a bi-objective two-machine re-etrant scheduling problem in which due to the strict due dates, outsourcing of the tardy jobs has been considered. We proposed a genetic-based algorithm with m-SCD dominance concept and adaptive generation size. We have shown that better estimation of non-dominated solutions could be achieved by comparing the results with NSGA-II regarding coverage, hypervolume and spacing metrics however, since in the proposed algorithm middle part of Pareto front was focused, less spread solutions were found. The results clearly indicate the ability of our proposed algorithm to find a good estimation of the middle part of the Pareto front. Testing this algorithm on large set of instances is in progress.

References

1. Danping, L., Lee, C.K.M.: A review of the research methodology for the re-entrant scheduling problem. International Journal of Production Research, 49 (8), 2221-2242 (2011).
2. Slotnick, S.A.: Order acceptance and scheduling: A taxonomy and review. European Journal of Operational Research, 212, 111 (2011).
3. Gonzalez, T., Sahni, S.: Flowshop and jobshop schedules: complexity and approximations. Operations Research, 26 (1), 36-52 (1978).
4. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transaction on Evolutionary Computation, 6(2), 182-97 (2002).
5. Sato, H., Aguirre H.E., Tanak, K.: Self-controlling dominance area of solutions in evolutionary many-objective optimization. Lecture Notes in Computer Science, 6457, 455-465 (2010).
6. Dugardin, F., Yalaoui, F., Amodeo, L.: New multi-objective method to solve reentrant hybrid flow shop scheduling problem. European Journal of Operational Research, 203, 22-31 (2010).
7. Choi, S.W., Kim, Y.D.: Minimizing total tardiness on a two-machine re-entrant flowshop. European Journal of Operational Research, 199, 375-384 (2009).
8. Tan, K.C., Lee, T.H., Khor, E.F.: Evolutionary algorithms with dynamic population size and local exploration for multiobjective optimization. IEEE Transactions on Evolutionary Computation, 5(6), 565-588 (2001).