Research Journal of Applied Sciences, Engineering and Technology 5(14): 3774-3776, 2013
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: September 15, 2012 Accepted: October 30, 2012 Published: April 20, 2013

Assessment of Possible Survey in Parallel Scheduling Using Non-Convex DEA

Hamed Hosseini
Department of Computer Engineering, Babol Branch, Islamic Azad University, Babol, Iran,
Tel.: 00989121793734

Abstract: An attempt has been investigated to show how applying DEA selected the most efficient among feasible solutions in parallel processing scheduling. One of the most important things between possible solutions is to determine the most efficient one. It will happen in NP-Complete problems when we are compelled to make heuristic algorithms to solve the problem. Maybe the heuristic proposed algorithm gives us some possible solutions. To reach the goal we are supposed to apply one decision maker. DEA model based on linear programming is widely used to make decision. This study will illustrate how DEA works on parallel scheduling problem and select the most efficient solution within which the objective is to minimize completion time and to minimize the number of tardy jobs.

Keywords: Linear programming, non-convex DEA, NP-complete

INTRODUCTION

Optimal finding is one of the most vital trends in computer science when we have some possible solutions and the objective is to find the most optimal. New technique of DEA was firstly introduced by Charnes *et al*. (1978). It is widely used as decision maker. DEA consumes appropriate inputs and outputs as Decision Making Units (DMU) and finally make the best decision. In this study we have used non-convex DEA because in the old version of DEA some efficient solution would be invisible. Non-convex DEA was extremely applied in retail banking and urban transit by Tulkens (1993) also it was applied to assess economic performance by Kuosmanen and Post (2003) and Pendharkar (2005) have used DEA for data processing. There are a lot of NP-Complete problems in computer science such as Traveler Salesperson Problem, Sum of Subset, Knapsack(0/1) and so on (Neopolitan and Naimipour, 2009; Cormen *et al*., 2009). Meanwhile, the problem of parallel scheduling is NP-Complete problem too. There is no polynomial time algorithm for them that is why we are needed to apply heuristic algorithm for them. When the heuristic algorithm runs, the output will be variety possible solutions. According to the goal function, DEA will select the most optimal solutions. This study implies that DEA defines the most optimal solution. The next part argues our problem and finally the important outcome will be illustrated in conclusion.

LITERATURE REVIEW

The method of DEA is applied for decision making. Assume that we have *n* decision making units (DMUs) and for each DMU we consume *p* parameters as input(s) and *q* parameters as output(s). For example DMU_k is (x_k, y_k) where, $X_k = (x_{1k}, \ldots, x_{pk})$ and $Y_k = (y_{1k}, \ldots, y_{qk})$. We use non-convex model because CCR model (Charnes *et al*., 1978) does not observe some inefficient units. Non-convex DEA model is defined for *k*th DMU as below (Deprins *et al*., 1984):

$$
\begin{aligned}
&\theta_k^* = \min \theta \\
&\quad s.t. \\
&-\sum_{j=1}^{n} x_{ij}\lambda_j + x_{ik}\theta \geq 0 \quad i = 1,\ldots,p \\
&\sum_{j=1}^{n} y_{rj}\lambda_j \geq y_{rk} \qquad r = 1,\ldots,q \\
&\lambda_j \geq 0, \lambda_j \in \{0,1\} \quad j = 1,\ldots,n
\end{aligned}
$$

In non-convex model λ_i is pertained to the set $\{0,1\}$. Abovementioned model will rank DMUs according to their performance. Application of non-convex DEA on our problem shall be explained.

ASSESSMENT OF POSSIBLE SOLUTIONS

For the sake of speed up computer and industrial engineers use parallel processing (Drozdowski and Dell'Olmo, 1999). Scheduling on multiprocessors in parallel is the problem of mapping jobs on processors with satisfying criteria like mean weighted flow time, completion time, mean weighted tardy jobs and so on. The notation α|β|δ was proposed by Blazewicz *et al*. (1996) and Veltman *et al*. (1990) to specify the problem characteristics. First part is used for processors' traits and the second part shows execution time and latter implies to criteria. Some important criteria

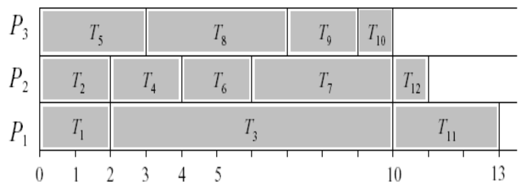

Fig. 1: Preemptive tasks schedule on three identical processors

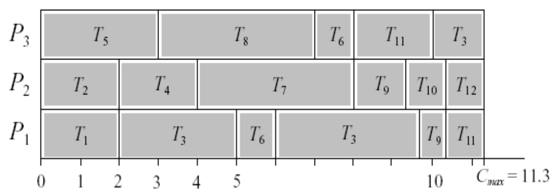

Fig. 2: Non-preemptive tasks schedule on three identical processors

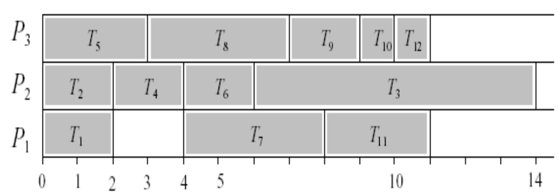

Fig. 3: Preemptive tasks schedule on three identical processors with deadlines

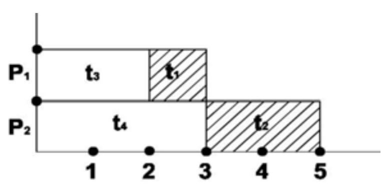

Fig. 4: Possible solution 1

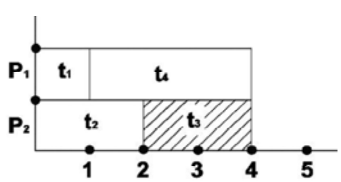

Fig. 5: Possible solution 2

$$C_{\max} = \max\{C_j \mid T_j \in \tau\}$$

$$\overline{F} = (\frac{1}{n})\sum F_j$$

$$L_{\max} = \max\{L_j \mid \quad T_j \in \tau\} \ \overline{D} = (\frac{1}{n})\sum D_j ,$$

and

$$\overline{U} = (\frac{1}{n})\sum U_j$$

Table 1: Schedule comparison with multi criteria

Criterion	Figure 1	Figure 2	Figure 3
C_{max}	13.000	11.333	14.000
$\overline{F}$	7.250	7.392	7.250
L_{max}	2.000	2.333	2.000
$\overline{D}$	0.167	0.360	0.167
$\overline{E}$	1.417	1.058	1.083
$\overline{U}$	0.250	0.333	0.167

Table 2: Task traits

Job No.	t_j	Due date (d_j)
T_1	1	2
T_2	1	2
T_3	2	3
T_4	3	4

Table 3: DEA specification for our problem

DMU no	Late Jobs (I_1)	Completion Time (I_2)	Number of jobs without delay (O_1)
Survey1(S1)	2	5	2
Survey2(S2)	1	4	3
Survey3(S3)	2	5	2

so that means total length, mean execution time, Max of Lateness, mean delay time, mean sum of tardy jobs respectively. Meanwhile, $\overline{E}$ is mean number of the early jobs? For example, consider we have 3 identical processors and 12 tasks; scheduling can be both preemptive and non-preemptive with different criteria. One feasible schedule was depicted as in Fig. 1 and processing time vector is (2, 2, 8, 2, 3, 2, 4, 4, 2, 1, 3,1) and scheduling is preemptive that is each task is executed from begin to end on specific processor without any interruption.

If interruption between tasks is legal, the schedule is non-preemptive and will be illustrated as Fig. 2.

Schedule in strict system such as real time system consumes deadline so that scheduling after that point of time is impossible and no profit gains. Consider (8, 2, 16, 4, 4, 8, 8, 8, 10, 8, 10 and 11) be deadline vector of tasks. Figure 3 illustrates preemptive schedule with deadlines.

Table 1 shows different criteria for schedules depicted on Fig. 1, 2 and 3. As can be seen lowest value in rows is colored with gray.

The problem of scheduling on multiprocessors is computationally hard to solve that is why for solution we have to use heuristic algorithm such as genetic algorithms. Our problem specification is as follows: Suppose that we have 2 identical processors with 4 tasks and interruption is prohibited. Objective is to minimize both C_{max} and number of the late jobs.

Table 2 shows jobs traits. Our applied genetic algorithm provides 3 surveys as can be seen in Fig. 4, 5 and 6. Moreover, the late jobs are hatched.

To determine the most efficient by DEA we clarify number of late jobs and execution time as inputs of DMU respectively and the number of jobs without lateness be an only output. We make DMU's specification in Table 3.

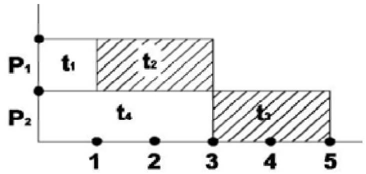

Fig. 6: Possible solution 3

According to non-convex DEA we are going to calculate the performance of unit1 as follows:

$$\theta_1^* = \min \theta$$
$$s.t.$$
$$-2\lambda_1 - \lambda_2 - 2\lambda_3 + 2\theta \geq 0$$
$$-5\lambda_1 - 4\lambda_2 - 5\lambda_3 + 5\theta \geq 0$$
$$2\lambda_1 + 3\lambda_2 + 2\lambda_3 \geq 2$$
$$\lambda_1, \lambda_2, \lambda_3 \in \{0,1\}$$

DEA-Solver is software application of non-convex DEA designed by Cooper *et al*. (2006). After solving aforementioned problem by DEA-Solver software, unit2 was determined as the reference set. It means survey2 is the optimal solution.

$$(\lambda_1^*, \lambda_2^*, \lambda_3^*, \theta_1^*) = (0,1,0,1)$$

CONCLUSION

The outcome shows that how non-convex DEA can be used as decision maker. In this study constructing some possible solutions by genetic algorithm were the input of non-convex DEA and finally this method got converged to survey2 as a reference set that can be seen in Fig. 5 as the most efficient solution.

REFERENCES

Blazewicz, J., K. Ecker, E. Pesch, G. Schmidt and J. Weglarz, 1996. Scheduling Computer and Manufacturing Processes. Springer, Heidelberg.

Charnes, A., W.W. Cooper and E. Rhodes, 1978. Measuring the efficiency of decision-making units. Eur. J. Oper. Res., 2: 429-444.

Cooper, W.W., L.M. Seiford and K. Tone, 2006. Introduction to Data Envelopment Analysis and its Uses: With DEA-Solver Software and References. Springer, New York.

Cormen, H.T., Ch. Leiserson, R. Rivest and C. Stein, 2009. Introduction to Algorithms. 3rd Edn., MIT Press, Cambridge, Mass.

Deprins, D., L. Simar and H. Tulkens, 1984. Measuring Labor-Efficiency in Post Offices, the Performance of Public Enterprises: Concepts and Measurements. North-Holland Publishing Co., Amsterdam.

Drozdowski, M. and P. Dell'Olmo, 1999. Scheduling multiprocessor tasks for mean flow time criterion. Comput. Sci. Res., 27(2000): 571-585.

Kuosmanen, T. and T. Post, 2003. Measuring economic efficiency with incomplete price information. Eur. J. Oper. Res., 168: 340-344.

Neopolitan, R. and K. Naeimipour, 2009. Foundation of Algorithm. 4th Edn., Harvard Business School Press.

Pendharkar, P.C., 2005. A data envelopment analysis-based approach for data preprocessing. IEEE T. Knowl. Data En., 17(10): 1379-1388.

Tulkens, H., 1993. On FDH efficiency analysis: Some methodological issues and applications to retail banking, Coutrs and urban transit. J. Prod. Anal., 4(1-2): 183-210.

Veltman, B., B.J. Lageweg and J.K. Lenstra, 1990. Multiprocessor scheduling with communications delays. Parallel Comput., 16: 173-82.