Single Machine Common Due Date Scheduling Problems Using Neural Network

Article *in* Jurnal Teknologi · June 2002

DOI: 10.11113/jt.v36.568

Shaharuddin Salleh
Universiti Teknologi Malaysia

Jurnal Teknologi, 36(C) Jun. 2002: 75–82
© Universiti Teknologi Malaysia

SINGLE MACHINE COMMON DUE DATE SCHEDULING PROBLEMS USING NEURAL NETWORK

ABDELAZIZ HAMAD, BAHROM SANUGI & SHAHARUDDIN SALLEH

Abstract. This paper presents an approach for scheduling under a common due date on static single machine problem based on artificial neural network. The objective is to sequence the jobs on the machine so that the total cost be minimized. This cost is composed of the total earliness and the total tardiness cost. Neural network is a suitable model in our study due to the fact that the problem is NP-hard. In our study, neural network has been proven to be effective and robust in generating optimal solutions to the problem.

Keywords: neural networks; static single machine; scheduling

Abstrak. Kertas ini membincangkan pendekatan rangkaian neural bagi masalah penjadualan dengan tarikh akhir sepunya pada satu kumpulan mesin statik yang tunggal. Objektif kajian ialah untuk menyusun kerja-kerja pada mesin supaya jumlah kos diminimumkan. Kos ini merupakan gubahan daripada jumlah kos awalan dan jumlah kos akhiran. Rangkaian neural adalah model yang sesuai dalam masalah ini kerana sifat masalah yang boleh dikategorikan sebagai NP-Sukar. Kajian ini membuktikan, rangkaian neural adalah efektif dan kukuh dalam menghasilkan penyelesaian hampir optimum kepada masalah ini.

Kata Kunci: rangkaian neural, mesin statik yang tunggal, penjadualan

1.0 INTRODUCTION

The study of earliness and tardiness penalties in scheduling models is a relatively recent area of inquiry. For many years, scheduling research focused on single performance measures, referred to as regular measures, that are non decreasing in job completion times. Most of the literatures deal with regular measures such as mean flowtime, mean lateness, percentage of jobs tardy, and mean tardiness. The mean tardiness criterion, in particular, has been a standard way of measuring conformance to due dates, although it ignores the consequences of jobs completing early. However, this emphasis has changed with the current interest in Just-In Time (JIT) production, which espouses the notion that earliness, as well as tardiness, should be discouraged [1]. In a JIT scheduling environment, jobs that complete early must be held in finished goods inventory until their due date, while jobs that complete after their due dates may cause a customer to shut down operations. Therefore, an ideal schedule is one in which all jobs finish on their assigned due dates. This can be translated to a scheduling objec-

[*] Department of Mathematics Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia.

tive in several ways. The most obvious objective is to minimize the deviation of job completion times around these due dates.

The concept of penalizing both earliness and tardiness has spawned a new and rapidly developing line of research in the scheduling field. Because the use of both earliness and tardiness penalties give rise to a nonregular performance measure, it has led to new methodological issues in the design of solution procedures. This paper presents a special case of the general Early/ Tardy (E/T) common due date problem when the earliness and tardiness are penalized at the same rate for all jobs. The next section introduces the problem statement. The neural network method and the scheduling of some small system are determined in the section after.

2.0 PROBLEM STATEMENT

The machine scheduling problem studied in this paper requires n independent jobs J_i (i =1,...,n) to be processed on a single machine with the following assumptions:

(i) All jobs are available for processing at time zero.
(ii) The single machine can process at most one job at a time.
(iii) No pre-emption is allowed.

Let s Schedule for the n jobs.
p_i Processing time required by job i on the machine.
d common due date.
c_i Time job i is completed after schedule s is started at time t = 0.
T_i Max $(0, c_i – d_i)$
E_i Max $(0, d_i – c_i)$
α_i Penalty per unit time for the earliness of job i.
β_i Penalty per unit time for the tardiness of job i.

An important special case in the family of E/T problems involves minimizing the sum of absolute deviations of the job completion times form a common due date. In particular, the objective function can be written as,

$$f(s) = \sum_{i=1}^{n} \left| c_i - d \right| = \sum_{i=1}^{n} (E_i + T_i) \tag{1}$$

with the understanding that $d_i = d$. When we write the objective function in this form, it is clear that earliness and tardiness are penalized at the same rate for all jobs ($\alpha_i = \beta_i$). The analysis of this problem is due to Kanet [5], Sundararaghavan and Ahmed [4], Hall [6], Chang [2] and Alidaee and Panwalkar [7]. In this paper we use neural network as heuristic method to solve this problem.

3.0 MULTI-LAYER NEURAL NETWORK FOR SINGLE MACHINE

An artificial neural network is a collection of highly interconnected processing units that has the ability to learn and store patterns as well as to generalize when presented with new patterns. The 'learnt' information is stored in the form of numerical values, called weights that are assigned to the connections between the processing units of the network. A neural network usually consists of an input layer, one or more hidden layers and an output layer. Before the network is trained, the weights are assigned small, randomly determined values. Through a training procedure such as backpropagation, the network's weights are modified incrementally until the network is deemed to have learnt the relationship. This type of learning is a supervised type of learning. When a pattern is applied at the input layer, the stimulus is fed forward until final outputs are calculated at the output layer. The network's outputs are compared with the desired result for the pattern considered and the errors are computed. These errors are then propagated backwards through the network as feedback to the preceding layers to determine the changes in the connection weights to minimize the errors. A series of such input-output training examples is presented repeatedly until the total sum of the squares of these errors is reduced to an acceptable minimum. At this point the network is considered 'trained'. Data presented at the input layer of a trained network will result in values from the output layer consistent with the relationship learnt by the network from the training examples. The neural network that is proposed for the single machine E/T common due date schedule problem is organized into three layers of processing units. There is an input layer of 10 units, a hidden layer, and an output layer that has a single unit. The number of units in the input and output layers is dictated by the specific representation adopted for the schedule problem [3]. In the proposed representation, the input layer contains the information describing the problem in the form of a vector of continuous values. The 10 input units are designed to contain the following information for each of the n jobs that have to be scheduled;

$$unit\ 1 = \frac{p_i}{M_p}, \tag{2.1}$$

$$unit\ = \frac{d}{100}, \tag{2.2}$$

$$unit\ 3 = \frac{SL_i}{M_{sl}}, \tag{2.3}$$

$$unit\ 4 = \frac{\alpha_i}{10.0}, \tag{2.4}$$

$$unit\ 5 = \frac{\beta_i}{10.0}, \tag{2.5}$$

$$unit\ 6 = \frac{\bar{p}}{M_p}, \tag{2.6}$$

$$unit\ 7 = 1, \tag{2.7}$$

$$unit\ 8 = \frac{S\bar{L}}{M_{sl}}, \tag{2.8}$$

$$unit\ 9 = \sqrt{\frac{\sum (p_i - \bar{p})^2}{n \times \bar{p}^2}}, \tag{2.9}$$

$$unit\ 10 = \sqrt{\frac{\sum (Sl_i - S\bar{L})^2}{n \times S\bar{L}^2}}, \tag{2.10}$$

where Sl_i slack for job $i = d - p_i$
M_p longest processing time among the n jobs = max (p_i)
M_{sl} largest slack for the n jobs = max (Sl_i), $i \in n$

The neural network is trained by presenting it with a predefined set of input and target output patterns. Each job is represented by a 10-input vector, which holds information particular to that job and in relation to the other jobs in the problem. The output unit assumes values that are in the range of 0.10-0.90, the magnitude being an indication of where the job represented at the input layer should desirably lie in the schedule. Low values suggest lead positions in the schedule; higher values indicate less priority and hence position towards the end of the schedule. The target associated with each input training pattern is a value that indicates the position occupied in the optimal schedule. The target value G_i for the job holding the *ith* position in the optimal schedule is determined as in equation (3);

$$G_i = \left\{ 0.1 + 0.8 \left(\frac{i-1}{n-1} \right) \right\}, \qquad i = 1, \ldots, n. \tag{3}$$

Equation (3) ensures that the n target values are distributed evenly between 0.1 – 0.9. The number of units in the hidden layer is selected by trial and error during the training phase. The final network for single machine common due date has 8 units in its hidden layer and 1 unit of output layer, therefore known as 10-8-1 network.

4.0 NUMERICAL EXAMPLES

4.1 Example 1

The trained neural network is used to find the schedule for minimizing the cost function of equation 1. Table 1 shows a 3-job single machine and common due date. The

Table 1 3-job scheduling problem

Job i	p_i	d	SI_i	α_i	β_i
1	8	15	7	1	1
2	10	15	5	1	1
3	4	15	11	1	1

3- jobs are converted first into their vector representation by using the set of equations (2.1-2.10). The result of this pre-processing stage is presented in Table 2 where the vectors V_1-V_3 represented job numbers 1-3, respectively. To solve the schedule problem, each vector is presented individually at the input layer of the neural network. A feed forward procedure of calculations generates a value that appears at the output unit for each of the three input vectors. The output computed by the neural network for each of the input vectors is given in the right most column of Table 2.

Table 2 Problem representation for the example described in Table 1
Input units

job											**out**
V_1	0.80	0.15	0.63	0.10	0.10	0.73	1.00	0.70	0.34	0.33	0.78
V_2	1.00	0.15	0.45	0.10	0.10	0.73	1.00	0.70	0.34	0.33	0.11
V_3	0.40	0.15	1.00	0.10	0.10	0.73	1.00	0.70	0.34	0.33	0.27

Scheduling the jobs in the order of the increasing output values results in the job schedule J_2-J_3-J_1 with total cost equal 13. The optimal schedule, on other hand, is J_2-J_3-J_1 with total cost equal 13. In this example the neural network scheduling give us optimal solution.

4.2 Example 2

In this example we use the same procedure as in example 1 but involved a 4-job scheduling problem (Table 3). The output computed by the neural network for each of the input vectors is given in the right most column of Table 4.

Scheduling the jobs in the order of the increasing output values results in the job schedule J_2-J_3-J_4-J_1 with total cost equal 26. The optimal schedule, on other hand, is J_1-J_4-J_3-J_2 with total cost equal 25. This implies that the neural network scheduling in this example give us near optimal solution.

Table 3 4-job scheduling problem

Job i	p_i	d	α_i	β_i	Sl_i
1	8	15	1	1	7
2	10	15	1	1	5
3	4	15	1	1	11
4	7	15	1	1	8

Table 4 Problem representation for the example described in Table 3
Input units

job											**out**
V_1	0.80	0.15	0.63	0.10	0.10	0.73	1.00	0.70	0.30	0.28	0.75
V_2	1.00	0.15	0.45	0.10	0.10	0.73	1.00	0.70	0.30	0.28	0.10
V_3	0.40	0.15	1.00	0.10	0.10	0.73	1.00	0.70	0.30	0.28	0.27
V_4	0.70	0.15	0.72	0.10	0.10	0.73	1.00	0.70	0.34	0.28	0.42

5.0 CONCLUSIONS

A single machine with common due date has been studied. The objective was to find an optimal scheduling that minimizes a cost function containing earliness costs and tardiness costs with symmetric penalties. A neural network model was developed to solve this problem. It was found that the multi layer neural network gave optimal solution for small system.

6.0 ACKNOWLEDGEMENT

The authors would like to thank Universiti Teknologi Malaysia (UTM) for providing the facilities to carry out this research.

REFERENCES

[1] Baker, K. S., and. G. D. Scudder. 1990. Sequencing with Earliness and Tardiness Penalties: A Review. *Op. Res*. 38: 22 – 36.

[2] Chang, T. C. E., and M. C. Gupta. 1989. Survey of Scheduling Research Involving due date Determination Decisions. *Euro. Journal Op. Res*. 38: 156 – 166.

[3] El-Bourni, A., S. Balakrishnan, and N. Popplewell. 2000. Sequencing Jobs on a Single Machine: A neural Network Approach. *Euro. Journal of Op. Res*. 126: 474 – 490.

[4] Sundararaghavan, P., and M. Ahmed. 1984. Minimizing the sum of Absolute lateness in Single and Multimachine Scheduling. *Naval Research*. Logistics. Quarterly 31: 325 – 333.

[5] Kanet, J. K. 1981. Minimizing the Average Deviation of Job Completion Times About a Common Due Date. *Naval Research*. Logistics. Quarterly, 28: 643 – 651.

[6] Hall, N. G. 1986. Single and Multi ple Processors Models for Minimizing Completion Time Variance. *Naval Research*. Logistics. Quarterly 33: 49 – 54.

[7] Alidaee, B., and S. S. Panwalkar. 1993. Single Stage Minimum Absolute Lateness Problem With a Common Due Date on Non-identical Machines. *Journal of Operational Research Society*. 44: 29 – 36.