IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 13, Issue 6 Ver. II (Nov. - Dec. 2017), PP 30-35
www.iosrjournals.org

A New Model of Neural Networks for Solving Identical Parallel Machine Scheduling With Distinct Due Date

Abdelaziz Hamad Elawad
Department of Mathematics, Faculty of Science, University of Al Baha
Saudi Arabia

Abstract: *In this paper the problem of Sequence the jobs on identical parallel machine with distinct due date has been studied, so that the total penalty cost to be minimized. This cost is composed of the total earliness and the total tardiness costs. We develop a new model system of neural networks base on unit's vector for inputs and output to solve our problems. For training purpose we use Mat Lap Software neural networks. After training the new model can solve similar problems. The performance of the new model system has been measured on numerical examples which taken from literature reviews. The results are very encouraging for further investigation.*

Key word: *Identical parallel machine, distinct due date, Earliness and Tardiness*

Date of Submission: 18-10-2017 Date of acceptance: 25-11-2017

I. Introduction

Due to the industrial significance of the just-in-time philosophy, due date problems, that means the scheduling problems where the due dates are given, have gained increasing attention in recent years. In such problems the jobs due dates are fixed by the customer. In this section, we discuss the case of scheduling *n* independent jobs on *m* parallel machines under given distinct due date. In our model for this section earliness is cosseted at the same rate for all jobs and so is tardiness, but the rates for earliness and tardiness different. In identical parallel machine each job requires a processing time, which are same for all machines. Schedules are assigned penalties, which are the sum of the costs related to earliness and tardiness of all jobs. By penalizing both the early and tardy completion of the jobs, costs related to inventory and customer satisfaction are recognized and taken into account. These costs are of a different nature. This is taken into consideration and incorporated into the model by allowing different weights for early and tardy completion. The objective is to schedule the jobs on the machines, so that the objective function is minimized. As the problem is NP-hard, the neural network approach is considered a good choice to provide solution to the problem.

Let n be the set of independent jobs $J_1, J_2,\ldots, J_n$ to be processed on m identical parallel machines. The following notation shall be used.

S schedule for the n jobs;
P_i processing time required by job i on the machine;
d_i distinct due date;
C_{ij} the completion time of J_i on machine j;
E_{ij} the earliness of J_i on machine j, which is equal to max $\{0, d - C_{ij}\}$;
T_{ij} the tardiness of J_i on machine j, which is equal to max $\{0, C_{ij} - d\}$;
P_2, P_3 the weights associated with earliness and tardiness, respectively.

The objective is to minimize the following cost function:

$$f(S)=\sum_{j=1}^{m}\sum_{i=1}^{n}(P_2E_{ij}+P_3T_{ij}). \qquad (1)$$

II. A Neural Network Model for Parallel Identical Machines

The neural network that is proposed for the parallel identical machine distinct due date schedule problem is organized into three layers of processing units. There is an input layer of 13 units, a hidden layer, and an output layer that has m units. The number of units in the input and output layers is dictated by the specific representation adopted for the parallel schedule problem. In the proposed representation, the input layer contains the information describing the problem in the form of a vector of continuous values. The 13 input units are designed to contain the following information for each of the n jobs that have to be schedule on parallel identical machines,

$$unit1 = \frac{p_i}{M_p}, \tag{2.1}$$

$$unit2 = \frac{d_i}{10}, \tag{2.2}$$

$$unit3 = \frac{SL_i}{M_{sl}}, \tag{2.3}$$

$$unit4 = \frac{P_2}{10}, \tag{2.4}$$

$$unit5 = \frac{P_3}{10} \tag{2.5}$$

$$unit6 = \frac{\bar{p}}{M_p}, \tag{2.6}$$

$$unit7 = \frac{\bar{d}}{M_d}, \tag{2.7}$$

$$unit8 = \frac{S\bar{L}}{M_{sl}}, \tag{2.8}$$

$$unit9 = \sqrt{\frac{\sum (p_i - \bar{p})^2}{n \times \bar{p}^2}}, \tag{2.9}$$

$$unit10 = \sqrt{\frac{\sum (d_i - \bar{d})}{n \times \bar{d}^2}} \tag{2.10}$$

$$unit11 = \sqrt{\frac{\sum (Sl_i - S\bar{L})^2}{n \times S\bar{L}^2}}, \tag{2.11}$$

$$unit12 = \frac{p_i}{\alpha \times 10} \tag{2.12}$$

$$unit13 = \frac{p_i}{\beta \times 10} \tag{2.13}$$

Where

P_i : Processing time of job i;

d : Common due date same for all jobs;

SL_i: Slack is the different between the due date and processing time $= d - P_i$;

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;

$$\overline{P} : \frac{\sum_{i=1}^{n} P_i}{n};$$

$$\mathrm{S}\overline{\mathrm{L}} : \frac{\sum_{i=1}^{n} SL_i}{\mathrm{n}}.$$

The neural network model is trained by using Mat Lap neural network software and by presenting it with a predefined set of input and target output patterns. Each job is represented by a 13-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 0.3 - 0.7 and zero for non-schedule job. 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_{ij} for the job holding the ith, position in the optimal schedule of machine j is determine as

III. Training Neural Network

We use the following data for training purpose, in this example we have 5 jobs and 3 machines.

Table(1) (5 Jobs and 3 machines)

Jobs	J_1	J_2	J_3	J_4	J_5
p_i	32/3	53/3	32/3	23/3	38/3
d_i	13	19	18	11	15
Sl_i	7\3	4\3	22/3	10\3	7\3
α	1	1	1	1	1
β	1	1	1	1	1

We transferd the above data of Table (1) by the 13 input units vectors then we get the data of Table (2), which is input data for neural networks.

$$G_{ij} = \begin{cases} 0.3 + 0.7\left(\dfrac{ij-1}{n-1}\right), & i = 1,\ldots,n \quad \text{and } j = 1,\ldots,n-1 \\ 0.00 & \text{non schedule.} \end{cases} \qquad (3)$$

Table(2): Inputs Vectors

0.6	1	0.6	0.43	0.72
0.13	0.19	0.18	0.11	0.15
0.32	0.18	1	0.5	0.32
0.1	0.1	0.1	0.1	0.1
0.1	0.1	0.1	0.1	0.1
0.05	0.05	0.05	0.05	0.05
0.68	1	0.95	0.58	0.79
0.32	0.18	1	0.5	0.32
0.11	0.11	0.11	0.11	0.11
0.11	0.11	0.11	0.11	0.11
0.06	0.06	0.06	0.06	0.06
0.2	0.2	0.2	0.2	0.2
0.62	0.62	0.62	0.62	0.62

Table (3) Desired output for training

M_1	0.00	0.00	0.415	0.312	0.00
M_2	0.00	0.348	0.00	0.00	0.00
M_3	0.542	0.00	0.00	0.00	0.679

The sequence of schedule for training example is J_4 - J_3 for Machine 1,
J_2 for Machine 2, and J_1 – J_5 for machine 3.

IV. Numerical Examples

Example (1)
In this example we have 7 jobs and 3 machines. We need to sequence these jobs on 3 machines so as the following cost function to be minimize.

This Example already solved by Krishna, N. and Samer. S (2015), they get total Cost equals 40.01. For our solution, we transferred the data of Table (4) to Table (5) by using the units Vectors (2.1) until (2.13).After that

$$f(S)=\sum_{j=1}^{m}\sum_{i=1}^{n}(\alpha E_{ij}+\beta T_{ij}). \tag{4}$$

we used Table (5) as inputs units for our Model Multi Layer neural network. Tables (6) illustrate our solution.

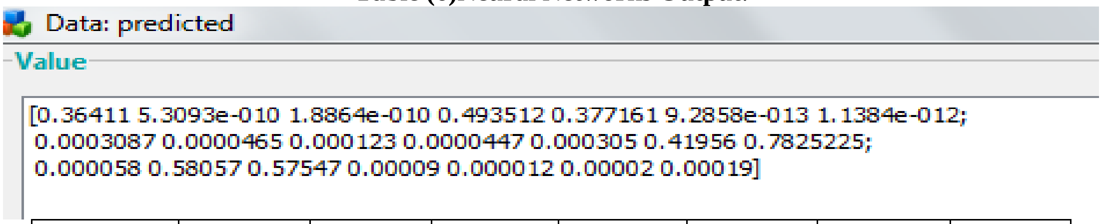
Table (4) (7 jobs and 3 machines)

job	1	2	3	4	5	6	7
p_i	26/3	23/3	35/3	29/3	23/3	38/3	29/3
d_i	12	18	19	16	10	13	15
Sl_i	10\3	31/3	4\3	19\3	7\3	1\3	16/3
α	1	1	1	1	1	1	1
β	1	1	1	1	1	1	1

Table (5) Input Units

0.68	0.61	0.92	0.76	0.61	1	0.76
0.12	0.18	0.19	0.16	0.1	0.13	0.15
0.32	1	0.13	0.61	0.23	0.03	0.52
0.1	0.1	0.1	0.1	0.1	0.1	0.1
0.1	0.1	0.1	0.1	0.1	0.1	0.1
0.76	0.76	0.76	0.76	0.76	0.76	0.76
0.77	0.77	0.77	0.77	0.77	0.77	0.77
0.41	0.41	0.41	0.41	0.41	0.41	0.41
0.87	0.77	0.12	0.97	0.77	0.13	0.97
0.87	0.77	0.12	0.97	0.77	0.13	0.97
0.18	0.18	0.18	0.18	0.18	0.18	0.18
0.2	0.2	0.2	0.2	0.2	0.2	0.2
0.76	0.76	0.76	0.76	0.76	0.76	0.76

Table (6)Neural Networks Output.



M_1	0.364	0	0	0.4935	0.377	0	0
M_2	0	0	0	0	0	0.4195	0.7825
M_3	0	0.58057	0.5757	0	0	0	0

Table (6) indicate the shedule sequence of the jobs. $\mathbf{J_1}$**-**$\mathbf{J_5}$ **-** $\mathbf{J_4}$ for $\mathbf{M_1}$, $\mathbf{J_6}$ **-** $\mathbf{J_7}$ for $\mathbf{M_2}$ and $\mathbf{J_3}$ **–** $\mathbf{J_2}$ for $\mathbf{M_3}$. Then the total cost of all machines is equals 19.67 + 7.67 + 8.67 = 36.01 which is better than their Krishna, N. and Samer. S (2015) cost, which is equal 40.01.

Example (2)

We use the same data of (Vahid and Mahdi, 2014) in the Table (7) below. They get total cost equals 6 units so as to compare between our schedules. In this example we have 8 Jobs and 3 machines.

Table (7) (8 jobs and 3 machines)

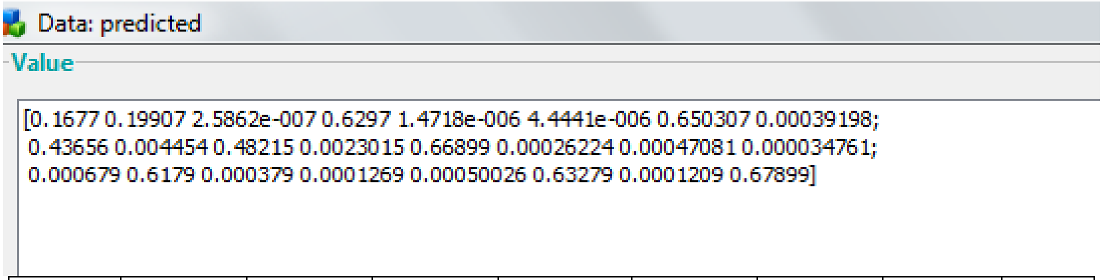
Jobs	1	2	3	4	5	6	7	8
P_i	4	6	5	7	5	6	4	6
d_i	5	5	11	6	13	13	11	20
α	0.5	1	1	1.25	1.5	1	1.5	0.5
β	0.5	0.5	1.25	0.5	0.5	1	3	0.5

We applied the same method for example 1. We transfer the data of table (7) to input units of table (8) by using the vectors units. After processing system of our model we get the outputs units illustrate by Table (9).

Table (8) Input units

0.57	0.86	0.71	1	0.71	0.86	0.57	0.86
0.1	0.1	0.11	0.1	0.13	0.13	0.11	0.2
0.2	-0.2	0.11	-0.2	0.16	0.14	0.14	0.27
0.05	0.1	0.1	0.13	0.15	0.1	0.15	0.5
0.05	0.05	0.13	0.05	0.05	0.1	0.3	0.5
0.77	0.77	0.77	0.77	0.77	0.77	0.77	0.77
0.53	0.53	0.53	0.53	0.53	0.53	0.53	0.53
0.37	0.37	0.37	0.37	0.37	0.37	0.37	0.37
0.8	0.6	0.5	0.56	0.33	0.6	0.13	0.12
0.8	0.12	0.4	0.14	1	0.6	0.13	0.12
0.18	0.18	0.18	0.18	0.18	0.18	0.18	0.18
0.46	0.46	0.46	0.46	0.46	0.46	0.46	0.46
0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.94

Table (9) Output Units of Neural Networks



M_1	0	0	0	0.6297	0	0	0.6503	0
M_2	0.4366	0	0.4822	0	0.6689	0	0	0
M_3	0	0.6179	0	0	0	0.6327	0	0.678

Table (9) indicate the shedule sequence of the jobs. $\mathbf{J_4}$-$\mathbf{J_7}$ for $\mathbf{M_1}$, $\mathbf{J_1 - J_3 - J_5}$ **for** $\mathbf{M_2}$ and $\mathbf{J_2 - J_6 - J_8}$ for $\mathbf{M_3}$.Then the total cost is equal 6, which is same of (Vahid and Mahadi, 2014) but with different in the sequence of schedule jobs on the machines. These indicate that this solution is unique.

V. Conclusion

The identical parallel machine with distinct due dates has been studied. The objective was to find an optimal scheduling of jobs on machines, which minimizes a total cost function containing earliness and tardiness costs with penalties. We developed a new model of multi layer neural network base on unit's vector variables to solve this problem. It was found that our model gets optimal or near optimal solution. Further investigation can be carried out for unrelated machine one.

References

[1] Makotoff ,E (2001)" parallel machines scheduling problems" . Asurvey Asia Pacific Jouranal of operation Research , 18,193-242
[2] B. Sanugi , A.Hamad and Salleh . (2001)" ANeural Network for distinct due date job scheduling problem on parallel identical machines ". Proc . Malaysia Science and Technology Congress 8-10:Melaka, Malaysia
[3] Gupta , J,N,D, Ruiz Torres , A, T and Webster (2003)" minimizing tardiness and number of tardy jobs on identical parallel machine". Journal of the Operations Research Society 54, 1263-1274
[4] Sunita and Singh ,T,(2010a)" Bi- objective in fuzzy scheduling on parallel machines ".Journal of mathematics and informatics 2,149-152
[5] Kewel Krishan (2015)" fuzzy bi – criteria scheduling on parallel machines involving weighted flow time and maximum tardiness ".Nailwal et al cogent mathematics 2,1019792
[6] Vahid and Mahdi Rajabi (2014)" unrelatedl parallel machine scheduling problem with sequence dependent setup time" Department of Industrial Engineering Amirkabir University of Technology 15916

Abdelaziz Hamad Elawad A New Model of Neural Networks for Solving Identical Parallel Machine Scheduling With Distinct Due Date. IOSR Journal of Mathematics (IOSR-JM) , vol. 13, no. 6, 2017, pp. 30-35.