Volume 3, Issue 5, May 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# Load Scheduling Algorithm Prediction for Multiple Tasks using Time Series Neural Network

**Saxena Sachin Kumar** [*]
*Department of IT,*
*COER, Uttarakhand, India*

**Kumar Dhaneshwar** [**]
*Department of IT,*
*COER, Uttarakhand, India*

**Singh Neha** [#],
*Department of CSE,*
*COER, Uttarakhand, India*

**Avasthi Vikas** [$]
*Department of MCA,*
*COER, Uttarakhand, India*

**Agarwal Gaurav** [@]
*Department of IT,*
*GRD-IMT, Uttarakhand, India*

***Abstract:*** ***Most scheduling problems have been demonstrated to calculate the time of scheduling algorithm problems. The neural network is commonly applied to obtain an optimal solution in various different scheduling applications, such as the traveling salesman problem (TSP), a typical discrete combinatorial problem. A time series algorithm in neural network is used for obtaining optimal schedules and the neural network is trained on these schedules. Knowledge is extracted from the trained network and past history data. The performance of this extracted rule set scheduling instances. The capability of the rule-based scheduler in providing dataset is highly used in various research works. This dynamic scheduling system should be seen as adaptive middle layer software, aware of current available resources and making the scheduling decisions using the past experience. It aims to optimize job specific parameters as well as the resource utilization. The scheduling system is able to dynamically calculation of finishing time and time consumed in scheduling the tasks of different parameters.***

***Keywords: Time Series Neural Network, Scheduling algorithm, Releasing time and Processing time.***

## I. INTRODUCTION

Finding and optimizing efficient job scheduling policies in large distributed systems, which evolve dynamically, is a challenging task. It requires the analysis of a large number of parameters describing the jobs and the time dependent state of the system. These jobs need random access to very large amounts of data, which are assumed to be organized and managed by distributed federations of OODB systems. Such a scheduling system may also help manage the way data are distributed among regional centers as a function of time, making it capable of providing useful information for the establishment and execution of data replication policies [3]. A large number of parameters, most of them time dependent, must be used for the job scheduling in large distributed systems [6]. The problem is even more difficult when not all of these parameters are correctly identified, or when the knowledge about the state of distributed system is incomplete or/and known with ascertain delay in the past. Time series neural network is more efficient to predict the data as well as provide the outputs using discrete input data of various tasks and parameters such as processing time, releasing time and deadline to finish the all the tasks. This work is beneficiary for the systems and machines which are autonomously schedule the things.

## II. METHODOLOGY

The deterministic job shop scheduling problem (JSSP) is one of the classical problems in scheduling literature [1]. JSSP consists of a finite set of n jobs to be processed on a finite set of m machines and is denoted as an n x m problem. The routing of a job is a predetermined sequence of operations. The methods used in this paper are Neural Network and time series prediction to predict the scheduling time which varies for different algorithm proposed earlier [2]. Any machine can easily determine using this technique that which scheduling algorithm is best in terms of money.

### II.a. JOB SCHEDULING

The knowledge base for the learning task was provided by the supervised learning solution to the job shop problem [5]. For this purpose, a prediction method for various kinds of inputs is chosen as the benchmark problem. This test instance has three tasks and processing time, releasing time and deadline. The data for the instance is shown in Table 1 using the following structure.

**Table 1. Attributes of each node**

| S.No. | Task Name | Processing Time | Releasing Time | Deadline |
|---|---|---|---|---|
| 1 | T1 | 6 | 9 | 15 |
| 2 | T2 | 8 | 15 | 18 |
| 3 | T3 | 11 | 18 | 20 |

 

The dependencies of tasks are also determined by figure 1 to show the appropriate releasing time and precedence constraints [8].

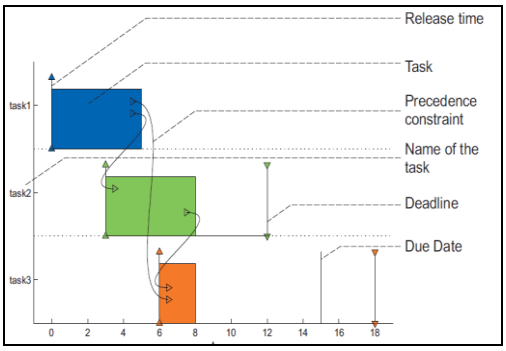


**Figure 1. Dependenccies of nodes**

## II.b. NEURAL NETWORK

This method is chosen to process on various kinds of inputs and to give the selection based outputs. Any machine or processor can easily determine or predict the future of the time taken by the scheduling algorithm.

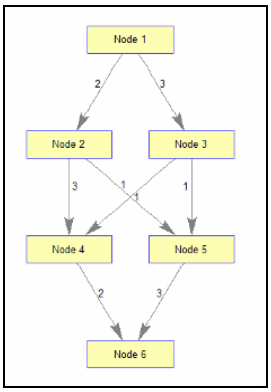


**Figure 2. Traversal of nodes along with weights in Direct Acyclic Graph**

The nodes can consist the dependencies through various resources therefore we need a direct acyclic graph to demonstrate the path of the system. In figure 3, the basic model of neural network is depicted and hidden input layers are illustrated to calculate the output of the next time series [9].

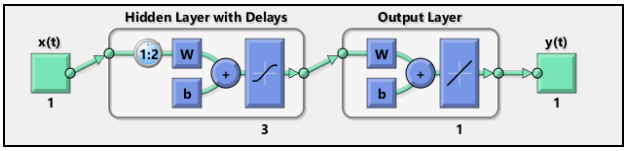


**Figure 3. Neural Network inputs and outputs**

## II.c. How Matlab Solves The Problem

MatLab R2012a is used to simulate the entire work because it can easily moderate the complexity of higher order problems [10]. The client mode is used to schedule the tasks whereas a MatLab worker process the system in MatLab in order to produce the results as depicted in figure 3.

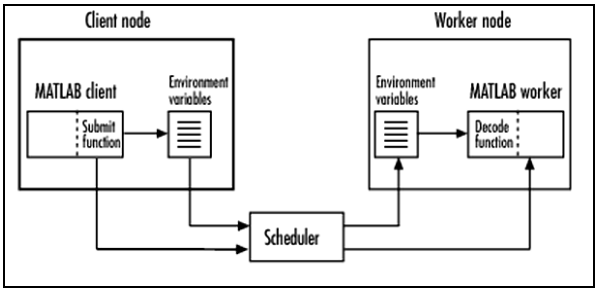


**Figure 4. How Matlab schedules the tasks**

 

## III. SCHEDULING ALGORITHM

### III.a. SCHEDULING WITH NEURAL NETWORK

There are attributes of Neural Network which is defined in table 2, time series back propagation network is used to predict the value because this method reduces the error in system by subtracting the output from target values [7].

**Table 2. Details of Neural Network Architecture used in Simulation**

| Number of Inputs | Number of Outputs | Numbers of Hidden Layers | Types of Prediction | Neural Network Types |
|---|---|---|---|---|
| Five | One | Three | Time Series | Back-Propagation |

Assignment of input features and target classes was done for each operation according to the classification scheme described in the previous subsection. Sample data for the classification task is shown in table 3.

### III.b. BRATLEY'S ALGORITHM

Bratley's algorithm, is proposed to analyze and solve problem, is algorithm which uses branch and bound method. Problem is from class NP-hard and finding best solution is based on backtracking in the tree of all solutions. Number of solutions is reduced by testing availabilty of schedule after adding each task [6].

```
>> T = taskset([2 1 2 2]);
>> T.ReleaseTime = ([4 1 1 0]);
>> T.Deadline = ([7 5 6 4]);
>> p = problem('1|rj,~dj|Cmax');
>> TS = bratley(T,p);
>> plot(TS);
```

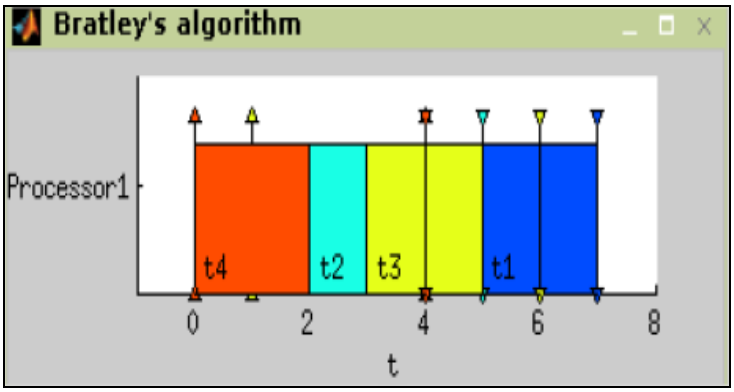


**Figure 5. Bratley's Algorithm Gantt chart**

### III.c. EARLIEST TIME FIRST ALGORITHM

Earliest Starting Time first (EST), intended to solve problem, is a strategy for LS algorithm in which the tasks are arranged in order of non-decreasing starting time before the application of List Scheduling algorithm. EST is implemented as an optional parameter to List Scheduling algorithm and it is able to solve problem.

```
>> t1=task('t1',3,10);
>> t2=task('t2',5,9);
>> t3=task('t3',5,7);
>> t4=task('t4',5,2);
>> t5=task('t5',9,0);
>> T = taskset([t1 t2 t3 t4 t5]);
>> p = problem('P|rj|sumCj');
>> TS = listsch(T,p,2,'EST');
>> plot(TS);
```

The outputs of this paper is defined in the scheduling the different tasks rather it is focus on time calculating to schedule the tasks [5]. The parameters of different tasks are so discrete that they raised the bar of complexity.

 

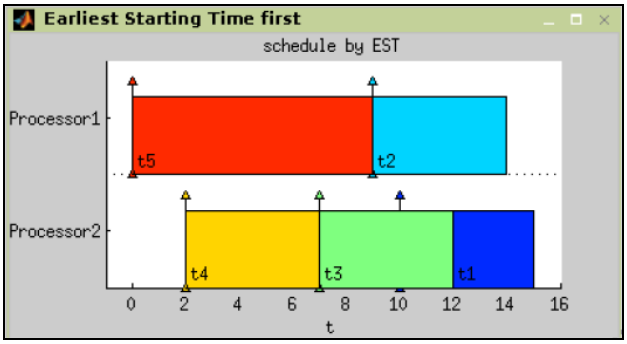


**Figure 6. Earliest Starting time first Algorithm Gantt chart**

The given commands are taken from MatLab command prompt and they are executed one by one to retrieve the Gantt chart to schedule the tasks. Each algorithm consists relevant time to finish their jobs; our aim is to determine the time to choose the minimum time scheduling algorithm [4]. This prediction will not only turn around the complexity of selection but also save the time in selection of different algorithm present in the market. The representation of charts explores the way in which these algorithms schedule the tasks of various attributes. Dependencies of the tasks are also elaborated in the graph to give the clear glimpse of resources available.

## IV. SIMULATION RESULTS

Entire system is simulated in MatLab R2012a and time series neural network toolbox to generate the outputs. Although, the attributes of the network is already defined in table 2, yet here once again we discuss about the results produced.

## V. RESULTS AND DISCUSSION

Neural Network predicts the value on the basis of supervised learning that is from the past experience and here are the results which were predicted by our network for each algorithm. One can easily estimated that McNaughton's consumes minimum time for the tasks along with attributes given in table 1.

**Table 3. Comparison of total scheduling time for different Algorithms**

| S.No. | Algorithm | Predicted Scheduling Time | Actual Scheduling Time |
|---|---|---|---|
| 1. | List Scheduling | 0.234 | 0.210 |
| 2. | *Bartley's Algorithm* | 0.431 | 0.388 |
| 3. | *Earliest Time First* | 0.532 | 0.489 |
| 4. | *McNaughton's* | 0.175 | 0.122 |

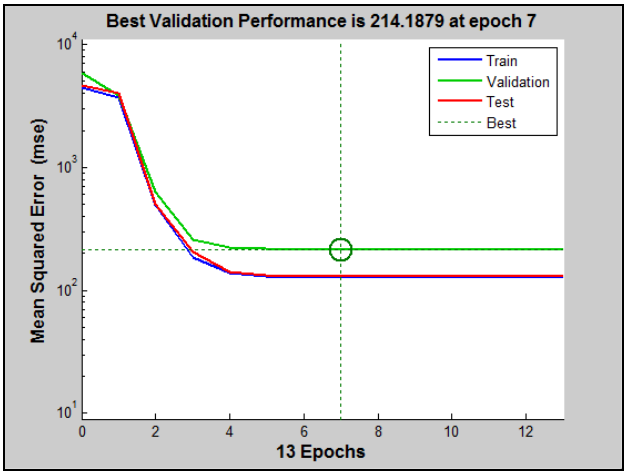


**Figure 7. Performance graph for the system**

 

Performance graph in figure 8 explains that best validation performance is 214.1879 after 7 epochs. Learning was halted after 13 epochs because provided input data was highly discrete and scattered.

## VI. CONCLUSION AND FUTURE WORK

The scheduling time of any algorithm is really important issue if one has large number of tasks, therefore we have moderate the method to select the scheduling algorithm among several scheduling methods available. Any machine or processor sometime unable to choose the algorithm to continue its work. This paper proposed the prediction of minimum time scheduling algorithm with help of several inputs and back propagation method in order to improve the error. The future work of the paper is remaining for the finishing time calculation for the jobs assign to one or many systems. The researchers are eagerly chasing the methods to predict the finishing time as earliest as possible.

## VII. ACKNOWLEDGEMENTS

This paper is supported by our mentor and guide Sachin Kumar Saxena, Member of Editor Team of International Journal of Advancements in Research & Technology (ISSN 2278-7763), IJOART Publication, USA. The website reference for it is given here to check http://www.ijoart.org/editorialTeam.shtml and College of Engineering Roorkee, Roorkee for providing the basic resources. The authors would like to thank Dr. Garima Krishna and Dhaneshwar Kumar in order to support in Matlab simulation work and compare this work to real environment.

## REFERENCES

[1] Yongbing ZHANG and Hisao KAMEDA, "Adaptive Bidding Load Balancing Algorithms in Heterogeneous Distributed Systems ", *IEEE 0-8186-5292-6/94, 1994.*

[2] Mahmood Al Bashir, Md. Zahidul Islam, Dr. A. K. M. Masud, " *Approach to Job-Shop Scheduling Problem Using Rule Extraction Neural Network Model* ", in *Global Journals Inc. (USA) ISSN: 0975-4172 Print ISSN: 0975-4350*

[3] Ruey-Maw Chen, Yueh-Min Huang, *Competitive neural network to solve scheduling problems,* 2001 Elsevier Science B.V., PII: S0925-2312(00)003441.

[4] Mathworks Pvt. Ltd., "Neuron Model and Network Architectures ", in *documentation* chap 2.

[5] Harvey B. Newman, Iosif C. Legrand , A Self-Organizing Neural Network for Job Scheduling in Distributed Systems, *Charles C. Lauristsen Laboratory of High Energy Physics California Institute of Technology, Pasadena, CA 91125, USA.*

[6] V. Venkatesa Kumar and K. Dinesh, Job Scheduling Using Fuzzy Neural Network Algorithm in Cloud Environment, Bonfring International Journal of Man Machine Interface, Vol. 2, No. 1, March 2012.

[7] Kothalil Gopalakrishnan Anilkumar and Thitipong Tanprasert , Neural Network Based Generalized Job-Shop Scheduler, Proceeding of the 2nd IMT-GT Regional Conference on Mathematics, statistics and Applications Universities Sains Malaysia, Penang, June 13-15, 2006.

[8] Seyed Hossein Kamali, Maysam Hedayati, Reza Shakerian and Saber Ghasempour, Performance Modeling of Distributed Load Balancing Algorithm Using Neural Networks, in International Journal of Computer Science and Telecommunications Volume 3, Issue 1, January 2012.

[9] Xinhua Jiang and Lina Zhang, An Application of Artificial Immune Algorithm for Load Balancing in VOD Cluster, in International Conference on Business Intelligence and Financial Engineering 2009.

[10] Dibyajyoti Guha, S.S. Pathak, Load Balancing using Past Information of Queue, in IEEE 978-1-4244-9477-4/11, 2011.

 