NEW DECISION MAKING TOOL USING NN SCHEDULING STRATEGY IN GRID HETEROGENEOUS SYSTEM

Ms. Smitha Jha  
Dr. D. K. Mallick  
Dr. R. K. Suri  
Asst. Professor, BIT Noida  
Associate Professor  
Director General,  
Email-s.jha@bitmesra.ac.in  
BIT Mesra, Ranchi  
Monad University, UP.

Abstract: Student’s employment in any company has been a big issue, as how the right student would be selected for right kind of job according to his potential. Different kinds of jobs can be taken as grid system. A tool is developed using Neural Network to schedule a particular student to the appropriate job in a company. Nprtool from MATLAB is used for the experiment. Training data is taken from BIT institute campus.

1. Introduction:

The Grid concept was firstly introduced by Foster and Kesselman [1, 2] who defined it as a distributed and decentralised computer environment composed of large number of heterogeneous resources which are managed by different owners such as companies, universities or another business or scientific organisations. Such resources may be of different type such as computational machines, data storage nodes, databases or scientific tools, etc. These resources are interconnected by high speed communication network. Most of these resources are built from relatively cheap components like regular PCs. Also other common hardware or existing infrastructure such as Internet is usually used. It is important to notice that the Grid environment
is changing dynamically in time mainly due to load of the system, available network bandwidth and also due
to other events such as resource or network failure or recovery.

The idea is that Grid technology managing this large and heterogeneous environment will allow an easy
access to its resources for various users, by means of allowing them to submit their jobs into the system,
guaranteeing them nontrivial Quality of Service (QoS) while hiding the complexity of the system itself by
providing powerful but simple interfaces for the end user of the Grid [1]. Moreover not only users but also
resource owners should be satisfied. In this case keeping the resource usage reasonably high is usually very
important. Therefore multi-objective criteria have to be met. To meet these goals sophisticated and
automated scheduling techniques should be applied. In a small scale system user can decide on his own
where the job will be executed but in a large scale system where thousands of CPUs, nodes and links are
available such assumption is not realistic. On the other hand, scheduling in such highly dynamic, distributed,
heterogeneous and decentralised environment is extremely difficult task if good performance, QoS or
robustness is required.

Current scheduling techniques applied in Grids are mostly based on the queueing systems of various types
which are designed with respect to specific needs of Grid technology. Since the Grids are built over existing
infrastructure their physical as well as logical topology is often hierarchical [3]. On the lower level Local
Resource Management Systems (LRMS) managing and scheduling jobs onto local resources of the
institution are applied. Nowadays systems like PBS [4], LSF [5], Sun Grid Engine [6] or Condor [7]
represent de facto standard solutions. On the higher level meta schedulers or so called brokers are used.
They usually manage different local sites that are each managed by some, often different, LRMS. Here a lot
of stand-alone solutions exists such as GridWay [8], Moab [9], GRMS [10] or Grid Service Broker [11].
Still, they are all using simple queue-based policies.

While single objectives can often be satisfied with some queue-based policy, complex objectives
such as response time, deadlines, resource utilisation etc., are hard to achieve by a queue-based solution,
especially for common user. Nowadays users seeking for better performance of their application are often
forced to cheat when looking for good performance of their applications. For example, if fast or immediate
response is needed user often executes his job by direct logging onto
specific machine and starting the job from the command line. In such situation the LRMS is bypassed which
can discriminate honest users and in general leads to inefficient performance of the whole system. Such
behaviour is often penalised by local administrators. Moreover, it is important to notice that in a long term
scope such situation can lead to both user's and administrator's frustration.

Neural Network model[12] has been designed to schedule tasks in Grid system using Neural network
approach. In this paper a slight variation of the above model has been introduced. It can be characterized as
classification problem using Neural Network. Here different kinds of jobs are taken as a grid system, as jobs are dynamic, distributed in nature. In this Neural Network model, different characteristics of student are taken as input. Different jobs are taken as output. Some backpropagation network is chosen to minimise the associated error. Network is trained with some set of data. On applying characteristics of any student, the output is able to tell the job to which the student should be employed. This develops a tool to describe the job to which any student would be employed.

2. Proposed tool

The proposed tool is able to decide about the job of any student, where he or she should be placed after the completion of course. This decision is made by looking the characteristics of the passing student. The tool is based on the Nprttool (Neural Network Pattern recognition tool) of MATLAB.

Structure of the tool:

| Input (A set of characteristics of student) | BackPropagation Network with hidden layer | Output (A set of different kinds of jobs) |

In the network the number of nodes at the input is taken in number equal to no. of characteristics of the student, say (c1, c2, c3, … cn). The no. of nodes at the output is equal to types of jobs say J1, J2, J3 … Jm). The no. of hidden nodes can vary according to performance requirement.

Passing 2012 batch of Management, Computer science is taken as training data. Characteristic of the student is taken as input of training set. Different types of jobs are taken as output in training set.

Each Characteristic of the student is evaluated by 3 faculties. The average is taken to minimize the biasing.

The Details of the characteristics of student (Input of training set) is described below

C1 - Technical skill (s/w skill)
C2-Communication skill

C3-Logical reasoning

C4-Language Proficiency

C5-Positive Attitude

C6-Participation in sports and cultural activities.

Here each faculty mark the student in one scale.

The details of types of jobs (Output of training set) is described below.

J1-HR

J2-IT

J3-Marketing

J4-Advertisement

J5-Call Center

J5-Teaching

J6-Sales.

The network is trained using training data (Table 1). At the output layer the one at output node depicts the type of job to which student should be employed.

Using this tool, any student can find its best choice of employment.
3. Experiment

In the experiment, training network is scaled conjugate gradient backpropagation network.

For experiment, data is collected from the database of 2012 passing batch. The no. of samples is equal to 126.

Here 70% students are taken for training. 15% students for Validation and 15% students for testing. No. of hidden neurons are taken as 20.

Here the Training set has two files

1. Studentinput.mat
2. Companyoutput.mat

<table>
<thead>
<tr>
<th>Roll No.</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C1</td>
<td>C2</td>
</tr>
<tr>
<td>4501/09</td>
<td>0.8</td>
<td>0.5</td>
</tr>
<tr>
<td>4502/09</td>
<td>0.5</td>
<td>0.8</td>
</tr>
<tr>
<td>So On</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table-1

Results:

Epoch: 61 iterations

Time: 3 second

Performance (Mean squared error): $4.00 \times 10^{-7}$
Gradient: 9.91 * 10^{-7}

Performance plots:

![Performance plots](image)

Best performance is 3.0104 * 10^{-6} at epoch 61.

4. Conclusion

This tool can be used to find the correct kind of employment for any student. The network could give better result if training data has diversity and more no. of data.

5. References:


