Systematic Viewpoint for Integrating Computational Resources by Using the Technique of PC Cluster

Kun-Lin Hsieh*
Department of Information Management & Research Group in Systemic and Theoretical Sciences
National Taitung University
684, Sec. 1, Chung Hua Rd., Taitung, Taiwan 950, R.O.C.

Cheng-Chang Jeng
Department of Education & Research Group in Systemic and Theoretical Sciences
National Taitung University
684, Sec. 1, Chung Hua Rd., Taitung, Taiwan 950, R.O.C.

Williams Tsai
Department of Information Management
National Taitung University
684, Sec. 1, Chung Hua Rd., Taitung, Taiwan 950, R.O.C.

Chun-Hung Lee
Department of Information Management
National Taitung University
684, Sec. 1, Chung Hua Rd., Taitung, Taiwan 950, R.O.C.

Abstract: - When we take a systematic view to screen out the general colleges or universities, we can find out a fact that many computational resources will be existed. During the competitive environment, how to sufficiently and efficiently utilize the computational resources under the consideration of the resource constrained had became an important job to most practitioners, especial for the general colleges or universities. In this study, we will introduce a case study in Taiwanese University owing the issue of computational resources integration based on the PC cluster environmental architecture and technique. This case is an application of clustering technique to DNA analysis. Not only the hardware/software architecture of PC cluster, but the clustering algorithm based on such cluster architecture was also proposed in this study to demonstrate the rationality and feasibility of real applications.

Key-Words: - PC cluster, Self-organizing feature map, computational resource.

1. Introduction

Personal Computers (PCs) had been viewed as the basic and necessary equipments at university. The fact that computational resources of CPUs for PCs will frequently denote idle state. Even though we execute several application programs on a single PC at the same time, the computational resources of CPUs still can keep a certain degree on the current operating system. Hence, such status will be viewed as one waste during the limitation of computational resource. Therefore, we can exchange to integrate those computational resources and release them to execute other jobs from a system viewpoint.

Besides, we had known that parallel computing can be performed well on a workstation or super computer, which are the devices with high price. However, the cost can not be accepted by most practitioners, especial for the resource constrained units. Many applications, e.g. pattern recognition or 3D computation and so on, will be gradually grow in the future for the real requirements from the research or studies. Hence, how to choose an approach with lower cost will be an
2. Background Introduction

2.1. PC cluster

If we can utilize PCs interconnected by an Ethernet/Fast Ethernet for distributed computing, the networked environment is called PC cluster (Sterling et al., 1999). Two kinds of PC cluster will be frequently met. The first one is a web/Internet cluster system and it will dispatch data into each node, that is, one service will be done simultaneously via many PCs. As for the second structure, it will dispatch all operations to each processor or node PC to complete the whole operation. For recent years, such technique also can be applied into PC. The multi-processors will be utilized to the technique of PC cluster. Hence, it will increase the performance of computation without high specification consideration. We can use the general PC to implement it. The important thing for PC cluster is that the node PC will be recommended with the same level to reduce the waiting time since performing computation (Sterling et al., 1999; Sloan, 2004).

To implement parallel programming in a PC cluster, the major issue is the distribution of information among PCs. In order to implement distributed computing, it is necessary to choose a programming interface that helps programmer to distribute messages among PCs. For a Windows-based PC cluster, there are several programming interfaces or software packages available. Message Passing Interface (MPI), the standard of message passing programming libraries, provides portable function calls to C, C++, and FORTRAN programmers. Since it is designed to be open sourced and used with homogeneous computer clusters, Windows-based MPI packages can be freely downloaded from the Internet. MPICH, a portable implementation of MPI, is available from Mathematics and Computer Science Division (MCS) (2003). MPICH is a modification and extension to MCS’s work. This alternative of MPICH also includes an enhanced version called NT-MPICH that improves the portability and performance of message passing to Windows NT/2000 environment (Bennmer, 2003).

2.2. Self-Organizing Feature Map (SOMNN)

The architecture form of the SOMNN network is based on the understanding that the representation of data features might assume the form of a self-organizing feature map that is geometrically organized as a grid or lattice. In the pure form, the SOMNN defines an “elastic net” of points (parameter, reference, or codebook vectors) that are fitted to the input data space to approximate its density function in an ordered way. The algorithm takes thus a set of N-dimensional objects as input and maps them onto nodes of a two-dimensional grid, and it will result in an orderly feature map (Kohonen, 1982; Kohonen, 1990).

The components in SOMNN are the input layer and the topological map, a layer of nodes topologically structured. Every input node is connected to every output node via a variable connection weight. A layer of two-dimensional array of competitive output nodes is used to form the feature map. Figure 1 depicts the architecture of a classic SOMNN. The lattice type of array can be defined to be square, rectangular, hexagonal, or even irregular. The SOMNN belongs to the category of the unsupervised competitive learning networks (Hinton, 1989; Kohonen, 1982; Kohonen, 1990; Hsieh et al., 2006). It is called competitive learning because there is a set of nodes that compete with one another to become active. In the SOMNN, the competitive learning means also that a number of nodes is comparing the same input data with their internal parameters, and the node with the best match (or it can be said as winner) is then tuning itself to that input, in addition the best matching node activates its topographical neighbors in the network to take part in tuning to the same input. More a node is distant from the winning node the learning is weaker. It is also

important issue for those practitioners and it will be the primary motivation of this study.

In this study, we demonstrated a successful case at a National University in Taiwan. This case is the clustering analysis for organism based on codon usage in DNA (Ghosh, 2000; Karlin Mrazek, 1996). And, it is a working research project at the Department of Information Management in Taitung University, Taiwan, R.O.C. No any financial support was provided and the cost will be viewed as the limitations of this working project. Then, this project team would like to apply PC cluster under a teaching laboratory into perform the application. Under such considerations, we will pay our attention to develop a rational and feasible architecture to achieve the analysis of organism clustering based on PC clusters architecture. Hence, the self-organizing feature map neural network (SOMNN) (Kohonen, 1984; Kohonen, 1982; Ye, 2003) will be the clustering technique to be chosen in this working project. The following sections will describe such successful case in detailed. The experimental architecture of PC clusters for hardware and software, and the clustering algorithm based on PC clusters will be also clearly described in this study.
called unsupervised learning because no information concerning the correct clusters is provided to the network during its training. Like any unsupervised clustering method, the SOMNN can be used to find clusters in the input data, and to identify an unknown data vector with one of the clusters. The detailed concept of SOMNN can be referred to Kohonen (1981, 1990).

3. Proposed Approach

In this section, we will clearly describe the hardware/software environment of PC clusters and the SOMNN algorithm based on PC clusters. According to our motivations of sufficiency utilization for computational resource, we choose Manufacturing Integration and Decision Supporting laboratory (MIDS) at Taitung University to the experimental place. At this Lab, we construct the PC clusters architecture with 5 PCs including one master and four nodes. The architecture diagram will be graphically depicted in Figure 1. As for the specifications of hardware/software for those 5 PCs, it will be given as follows:

1. CPU: Intel Pentium 4 3.2Ghz
2. RAM: 512MB DDRII RAM
3. Capacity: 80GB HDD
4. OS: Debian Linux Sarge(v3.10)

In order to achieve the Grid computing mechanism, the following design concept will be taken into considerations:

1. Firstly, five partitions will be formed for the master PC. The command will be listed as (/cluster/server -> master working area). The main process, client sub-process and source data will be included in this area and total size is about 20GB;
2. As for four nodes PCs, it will be listed as (/cluster/node1 ~ node 4 -> slave working area). Basically, this area is empty before the main process is “enable”. And the total size is given as about 5GB;
3. Next, those four nodes PCs will mount to the master working area by using NFS. It will provide a capacity about 20GB * 5 = 100GB. That is, the working area (/cluster/server) is shared for those four nodes PC.
4. The mechanism of mounting for each node PC, e.g. (/cluster/node) of the node 1 can mount to (/cluster/node1) of master, will provide master the authority to use the section of each node.

4. Implantation and Result

The researching group took data from GONOME ALTAS DATABASE (it can be obtained from http://www.cbs.Dtu.dk/services/GenomeAtlas/) to perform the analysis. At first, six bacteria with the full DNA sequence data will be chosen and the related information will be given as Table 1. A few and definite data will be used to verify the accuracy of the designed algorithm and architecture. The project team made a pre-processing for those DNA sequence to obtain the evaluation index, which is called as codon usage (Ghosh, 2000; Karlin Mrazek, 1996). All bacteria will include three amino acids (LEU, SER, ARG) data with six codon value, restated, totally have eighteen value to be considered for each bacteria. Basically, those six bacteria will be grouped into two clusters (Helgason et al., 2000; TIGR, 2005): (A, B, C) and (D, E, F). That is, we meet a clustering problem with a structure of 6*18.

Next, we will apply the designed SOMNN-PC cluster algorithm to perform the clustering analysis. The operation diagram can be referred to Figure 2. Herein, Figure 2 will represent the setting windows to set the number of nodes; Figure 3 will represent the final clustering result derived from SOMNN-PC cluster. From those operation Figures, we can find out that those bacteria can be clustered into three clusters. The first cluster includes A, B, C; the second cluster includes D, E and F. Such clustering result is the same as the actual result. Restate, the rationality, feasibility and accuracy of our proposed SOMNN-PC cluster algorithm can be verified.
Table 1. The related data about the eighteen bacteria.

<table>
<thead>
<tr>
<th>No.</th>
<th>Organism Label</th>
<th>Accession No.</th>
<th>Bases(bps) No.</th>
<th>Taxo. ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bacillus anthracis str. Ames</td>
<td>AE016879</td>
<td>5227293</td>
<td>198094</td>
</tr>
<tr>
<td>2</td>
<td>Bacillus anthracis str. 'Ames Ancestor'</td>
<td>AE017334</td>
<td>5227419</td>
<td>261594</td>
</tr>
<tr>
<td>3</td>
<td>Bacillus anthracis str. Sterne</td>
<td>AE017225</td>
<td>5228663</td>
<td>260799</td>
</tr>
<tr>
<td>4</td>
<td>Bacillus cereus ATCC 10987</td>
<td>AE017194</td>
<td>5224283</td>
<td>222523</td>
</tr>
<tr>
<td>5</td>
<td>Bacillus cereus ATCC 14579</td>
<td>AE016875</td>
<td>5411809</td>
<td>226900</td>
</tr>
<tr>
<td>6</td>
<td>Bacillus cereus ZK F</td>
<td>CP000001</td>
<td>5300915</td>
<td>288681</td>
</tr>
</tbody>
</table>

Next, the researching group extend the data to about fifteen bacteria by referring to the study of Hsieh et al. (2006). And the clustering result will be given as Figure 5. We obtained the clustering result as five clusters: (No.9, No.10, No. 11), (No.12, No.13, No.14, No. 15), (No.4, No.8), (No.2, No.3, No.5, No.6, No. 7) and (No.1). And the results will be very close to the result from Hsieh et al. (2006). After reviewing the comparison, we can find out the No. 1, No. 4, No. 8 is divided into two clusters: (No. 1) and (No. 4, No. 8), however, the result obtained from Hsieh et al. (2006) is grouped into one cluster. The other cluster results are the same. According to the concluding remarks from Hsieh et al. (2006), we can find out that five clusters or four clusters can be also viewed as a rational choice. Although it will denote a slight different result, the rationality and feasibility of the proposed SOMNN-PC cluster algorithm can still be demonstrated. After making the necessary discussion, we recognize that the initial setting for the weight value of SOMNN, the enhanced algorithm in SOMNN and the chosen parameters (i.e. the dimensions) inputting to SOMNN will be possible factors to lead the different clustering result. However, no significant difference between those two analysis results is met. Hence, the rationality and feasibility still can be verified.

5. Concluding Remarks

In this study, we introduce a case study based on the integration of computational resources from the systematic viewpoint. The primary concept is based on PC cluster technique to achieve the clustering analysis. In this case, a rational and feasible algorithm of SOMNN-PC cluster is to propose at the laboratory environment of university. Except for the efficiency (i.e. processing time) of PC cluster architecture can be enhanced, the function of clustering analysis can also be made. This practice can be viewed as an application reference for the issue of PC cluster architecture. However, from the viewpoint of optimum utilization of system, the efficiency of SOMNN algorithm based on PC cluster still have the chance to enhance or modify, e.g. including MPI into re-coding our algorithm. Hence, it will be our subsequent research in the future. Besides, how to apply the proposed PC cluster architecture into the case with large size, e.g. directly using the DNA sequence code to perform the clustering analysis, will also be a meaningful research in the future.

Acknowledgment
The authors would like to thanks the financial support from the National Science Council at Taiwan, R.O.C. with contract No. NSC 94-2416-H-143-001.

References