

Explore on Parallel Computing Model of BSP in the NOWs Adapt Environment

¹M. Srinivas, ²R. Ravi Kumar, ³G. Yedukondalu

^{1,2,3}Dept. of CSE, SBIT, Khammam, AP, India

Abstract

The paper analyze the characteristic of Parallel Computing Model of BSP and NOWs, Explore on Parallel Computing Model of BSP in the NOWs fit Environment. Indicate: some algorithm that designed rationalization parallel computing, acquire an approximately linear accelerated. With the parallel computing algorithm base on the NOWs of linear programming normal improve on simplex method obtain had best result to validate the conclusion.

Keyword

BSP model, NOWs, Parallel computing, Cost formula, Accelerated ratio, Linear.

I. Introduction

The development of the modem computer, people has done their best to increase the rate of the computer performance. But with the advancement of science, the demand of the performance of the computer is how to measure. In many areas, such as engineering design, autoimmunization, reconnoitering, medicine, military and basic theory research, there are higher demands on computation. But the traditional system structure of the computer and the limits of the physical part of parts of apparatuses. In the endeavor of developing the computer of new generation, the same characteristic is the use of parallel technology. That is to increase the number of operation at the same time interval, and it is called parallel treatment technique. Computers designed for parallel treatment are called parallel computers; the solution on the parallel computer is called parallel computation; the algorithms implemented to solve problem on parallel computers are called parallel algorithms. But the expense of the development and the cost are high. The ratio of performance to price of MPP grew rapidly since 1980's system compared with that of Super Computer. At present the highest operating rate of MF'P system can reach the scale of trillion. But MPP needs lots of initial investment, because the implemental cycle is long, and CPUs of the nodes could not be upgraded in time. Few programmers could possess this environment, so there are few software's supporting MPP. As a result, MPP systems cost a lot and are short and flexibility. Furthermore it is difficult to be upgraded and maintained. Because of the limit of structure, the number of CPUs in the second class SMP could not be too large.

In recent years, people show great interest in using a parallel Cluster which is made up of cheap, powerful work station or advanced pc groups. Cluster has become the hotspot and main stream and gained general emphasis domestically and abroad due to its apparent features. Little investment risk, good scalability, inherits recent software's and hardware's resource, short exploring period are among its apparent features.

A lot of works must be done to implement in Parallel programming designing: how to use parallel computer and to improve its efficiency in farthest; the application program base on parallel programming environment is very easy migrate from a parallel system to another's. Current some parallel programming environment such as PVM and MPI presents in intemational, bring huge benefit in parallel programming migrate.

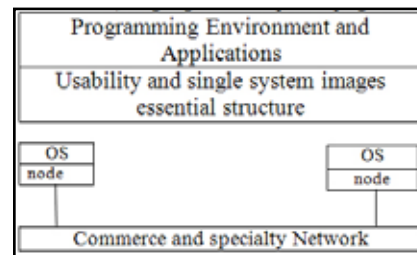


Fig. 1: A typical NOWs construction

The paper presents BSP (Bulk Synchronous Parallel) parallel algorithm model is a kind of independent in parallel system construction and is a simple validity parallel programming model, is also a bridge for parallel programming language with the parallel system construction. BSP is an abstract machine models and its high construction make its can predict the overall communication time and this is what a lot of another parallel model do not have. But because of different parallel computer of the ability that parallel construction and abstract machine operation exists margin, make the BSP parallel process at in a specific way parallel compute is not necessarily efficiently of, therefore, must aim at to NOWs the characteristics, study to lift the method of the high performance.

II. BSP Model in NOWs Environment

The NOWs technique for of the construction and adopt from essentially with the LAN's difference also not very, how key consist in make a lot of the node be in conjunction with the work, and together complete to compute the mission.

The BSP model, which is proposed by LGValiant in 1990, is used to, design parallel algorithms. Due to the effort of the Computing Laboratory of Oxford and other research organizations, it has been widely applied in various scientific fields such as signal processing, molecular dynamics and fluid dynamics. Because the algorithms designed with this model are scalable, portable and predictable, it provides a new foundation for the development of scalable parallel computing systems. The BSP model views a parallel machine as a set of these three elements:

- A. A set of processor-memory pairs.
- B. A global communication network that delivers messages in a point-to-point manner among the processors.
- C. A mechanism for globally synchronizing all processors by means of a barrier. A BSP program consists of a sequence of supersteps: Every superstep can be divided into three ordered phases:
 1. Simultaneous local computation in each processor, using only values stored in its memory.
 2. Communication among processors for exchanging information.
 3. Barrier synchronization for ensuring that communication actions are completed successfully.

The modular structure of the BSP model is conducive to realize algorithms and facilitates computing the cost. The cost of a sequence of supersteps is simply the sum of the costs of the separated supersteps. As an individual superstep contains

the three ordered phases of computation, communication and synchronization, it is natural to express the cost of a superstep by a formula, which is:

$$C = (\max(w_i) + \max(h_i g) + 1) \cdot N$$

$$0 \leq i \leq p-1 \quad 0 \leq i \leq p-1$$

where, p is the amount of processors, which are numbered from 0 to $p-1$. Intuitively, the cost of a superstep is the execution time of the processor that performs the largest volume of data ($\max(w_i)$), plus the communication time of the processor sending and receiving the largest volume of data ($\max(h_i g)$), plus a constant cost that arises from the banner synchronization. h_i is defined as $h_i = \max(h_{in_i}, h_{out_i})$, where h_{in_i} , h_{out_i} represent the volume of the data received and sent by processor i respectively. The parameter g reflects the permeability of the network and can be regarded as the time of transferring a unit data. It is relevant with the parallel architecture and the communication pattern. N is the number of supersteps.

The NOWs usually use general local network technology, so the capability of computing and network communication of the nodes is extremely unbalanced. the performance of NOWs is greatly limited by the network although some nodes of NOWs can run better than single node of parallel computer. Considering those points, when design parallel programming. we should pay a attention to those:

- (i). Task distributes balance. BSP program is SPMD structure, the BSP cost formula indicial: local computing cost in a superstep be decided by that node that computing time is the longest, so the algorithm design must ensured which node task approximately is propinquity in a superstep.
- (ii). Coarse grain task distributes, endeavor to reduce the communication of the each other node: because of speed is slower of the communication of NOWs, overabundance communication fall the efficiency of NOWs points:

III. Parallel Algorithm Effect Analyze In Now

The essential aim of parallel algorithm is reduce the time of computer deal with task and improve on the efficiency of which node in NOWs:

Assume a task Q resolved using an 'parallel algorithm in a NOWs with the amount of P processors run time is T_p , the same task Q resolved in computer of with a single processor run time is T_1 and the accelerate ratio is:

$$Sp = \frac{T_1}{T_p} \quad (1)$$

The Formula (1) clear presents $Sp \propto \frac{1}{T_p}$, the Sp , is more great and indicate the parallel algorithm running time is shorter and is more efficiency when amount of P is fixed. But Sp , related with the characteristic of parallel algorithm and also related with the amount of p in parallel computer. So more clearly indicate the parallel algorithm effect with formula:

$$Ep = \frac{Sp}{P} \quad (3)$$

Because commonly $T_p \geq \frac{T_1}{P}$ so has $Ep \leq 1$ the Ep more approach more excellence effect. In fact, Ep less than 1 because communication time cost, synchronization time cost etc.,

The paper with linear programming normal improves on parallel algorithm simplex method parallel to validate the conclusion. The simplex method is improved on two class simplex method. The parallel algorithm based the in series algorithm. The parallel algorithm processing problems resolved the liner programming normal:

$$\begin{cases} \min f = cx \\ s.t \quad ax = b \\ x \geq 0 \end{cases}$$

The fig. 2, is a test result in NOWs, the Detailed methods are select difference scale: 100*101, 200*201, 300*301, 400*401, 500*501 normal problem, the amount of CPU is 1, 2, 4, 8.

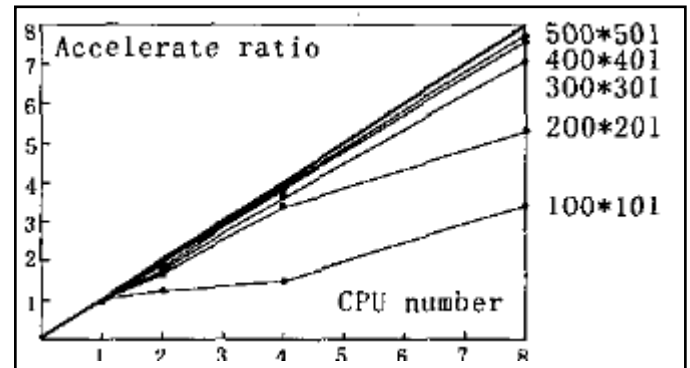


Fig. 2: The accelerate ratio related with amount of CPU

The fig. to validate the conclusion: more the amount of CPU and the great scale problem, the accelerate ratio is increase and approximately linear accelerated. So Liner programming simplex method adapt parallel algorithm in NOWs.

IV. Conclusion

The paper analyze the adapt environment of Parallel Computing Model of BSP in NOWs, Indicate: some algorithm that designed rationalization parallel computing, acquire an approximately linear accelerated.

References

- [1] Jonathan M.D. Hill, Bill McColl, Dan CStefanescu, Mark WGoudreau, Kevin Lang, Satish B.Rao, Torsten Suel, Thanasis Tsantilas, Rob Bisseling, BSPlib, "The BSP Programming Library", Technical Report PRG-TR-29-97, Oxford University Computing Laboratory, May, 1997.
- [2] Jonathan M.D. Hill, David Skillicorn, "Lessons learned from implementing BSP", Journal of future Generation Computer Systems, 1998.
- [3] Kai H u g, Zhiwei Xu, "Scalable Parallel Computing: Technology, Architecture, Programming", pp. 18-22, Chinese Machine Press, 1999.
- [4] Andrew S. Tanenbaum, "Computer Network", (third edition), Prentice Hall International, Inc.



M. Srinivas received his M.Sc degree in Computer Science from Andhra University, Vizag and his M.Tech in PARALLEL COMPUTING in the year 2010 from JNT University, Hyderabad. At present working as Assistant Professor in CSE department at Swarna Bharathi Institute of Science and Technology, Khammam, AP. His research interests includes Parallel Computing, Grid Computing and MANETs.



G. YeduKondalu received his MCA degree in Computer Science in the year 2006 from Kakatiya University, Warangal and M.Tech in Computer Science from JNTUH in 2009-11. At present working as Assistant Professor in CSE department at Swarana Bharathi Institute of Science and Technology, Khammam, AP, His research interest includes Network Security and MANETs.



R. Ravikumar received his MCA degree in the year 2001 from Kakatiya University, Warangal and his M.Tech(CSE) from JNT University, Hyderabad in 2008-10. At present working as Assistant Professor in CSE department at Swarana Bharathi Institute of Science and Technology, Khammam, AP, His research interest includes Network Security, Parallel Algorithms and MANETs.