DYNAMIC TASK PARTITIONING MODEL IN PARALLEL COMPUTING

Javed Ali[1] (Research Scholar) and Rafiqul Zaman Khan[2](Associate Professor)

javedaligs@gmail.com, rzk32@yahoo.co.in
Department of Computer Science,
Aligarh Muslim University, Aligarh, India.

ABSTRACT

Parallel computing systems compose task partitioning strategies in a true multiprocessing manner. Such systems share the algorithm and processing unit as computing resources which leads to highly inter process communications capabilities. The main part of the proposed algorithm is resource management unit which performs task partitioning and co-scheduling .In this paper, we present a technique for integrated task partitioning and co-scheduling on the privately owned network. We focus on real-time and non preemptive systems. A large variety of experiments have been conducted on the proposed algorithm using synthetic and real tasks. Goal of computation model is to provide a realistic representation of the costs of programming The results show the benefit of the task partitioning. The main characteristics of our method are optimal scheduling and strong link between partitioning, scheduling and communication. Some important models for task partitioning are also discussed in the paper. We target the algorithm for task partitioning which improve the inter process communication between the tasks and use the recourses of the system in the efficient manner. The proposed algorithm contributes the inter-process communication cost minimization amongst the executing processes.

KEYWORDS: *Criteria, Communication, Partitioning, Computation, Cluster.*

1. INTRODUCTION

Parallel computing is used to solve the large problems in the efficient manner. The scheduling techniques we discuss might be used by an algorithm to optimize the code that comes out of parallelizing algorithms. Thread can be used for task migration dynamically [1].The algorithm would produce fragments of sequential code, and the optimizer would schedule these specks such that the program runs in the shortest time. Another use of these techniques is in the design of high-performance computing systems. A researcher might want to construct a parallel algorithm that runs in the shortest time possible on some arbitrary computing system which is used to increase the efficiency and decreases the turnaround time. Parallel computing systems are implemented upon platform comprise of the heterogeneous platforms comprise the different kinds of units, such as CPUs, graphics co-processors, etc. An algorithm is constructed to solve the problem according to the processing capability of the machines used on the cluster and mode of

Sundarapandian et al. (Eds): CoNeCo,WiMo, NLP, CRYPSIS, ICAIT, ICDIP, ITCSE, CS & IT 07,
pp. 279–284, 2012. © CS & IT-CSCP 2012 DOI : 10.5121/csit.2012.2425

communication amongst the processing tasks [10]. The communication factor is the highly important feature to solve the problem of task partitioning in the distributed systems. A computer cluster is a group of computers working together closely in such a manner that it's treated as a single computer. Cluster is always used to improve the performance and availability over that of a single computer. Task partitioning is achieved by linking the computers closely to each other as a single implicit computer. The large tasks partitioned in the various tasks by the algorithms to improve the productivity and adaptability of the systems. A cluster is used to improve the scientific calculation capabilities of the distributed system [2]. The process division is a function that divides the process into the number of processes or threads. Thread distribution distributes threads proportionally according to the need, among the several machines in the cluster network [chandu10].Thread is a function which execute on the different nodes independently so communication cost problem is not considerable[3]. Some important model [4] for task partitioning in parallel computing system are: PRAM ,BSP etc.

1.1 PRAM MODEL

It's a robust design paradigm provider. PRAM composed of P processors, each with its own unmodifiable program. A single shared memory composed of a sequence of words, each capable of containing an arbitrary integer [5]. PRAM model is an extension of the familiar RAM model of sequential computation that is used in algorithm analysis. It consists of a read-only input tape and a write-only output tape. Each instruction in the instruction stream is carried out by all processors simultaneously and requires unit time, reckless of the number of processors. Parallel Random Access Machine (pram) model of computation consists of a number of processors operating in lock-step and communicating by reading and writing locations in a shared memory in efficient and systematic manner[13].In its model each processor has a flag that controls whether it is active in the execution of an instruction or not. Inactive processors do not participate in the execution of instructions.

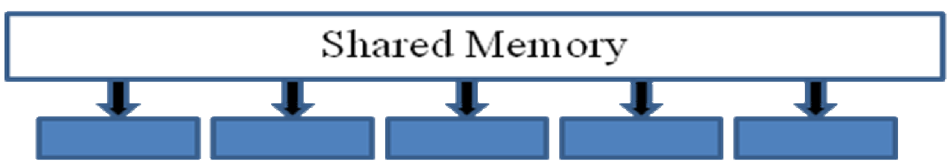

Figure 1.PRAM Model Shared Memory

The processor id can be used to distinguish processor behavior while executing the common program. The operation of a synchronous PRAM can result in simultaneous access by multiple processors to the same location in shared memory. The highest processing power of this model can be used by using Concurrent Read Concurrent Write (CRCW) operation. It's a baseline model of concurrency and explicit model which specify operations at each step[11]. It allows both concurrent reads and concurrent writes to shared memory locations. Many algorithms for other models (such as the network model) can be derived directly from PRAM algorithms[12]. Classification of the PRAM model:

1. In the Common CRCW PRAM, all the processors must write the same value.
2. In the Arbitrary CRCW PRAM, one of the processors arbitrarily succeeds in writing.

3. In the Priority CRCW PRAM, processors have priorities associated with them and the highest priority processor succeeds in writing.

2. PROPOSED MODEL FOR TASK PARTITIONING IN DYNAMIC SCHEDULING

Task partitioning strategy in parallel computing system is the key factor to decide the efficiency, speedup of the parallel computing systems. The process is partitioned into the subtasks where the size of the task is determined by the run time performance of the each server [9]. In this way assign no. of tasks will be proportional to the performance of the server participate the distributed computing system. The inter process communication cost amongst the task is very important factor which is used to improve the performance of the system [6]. The scheduler schedules the tasks and analyzes the performance of the system. The inter processes communication cost estimation criteria in the proposed model is the key factor for the enhancement of the speed up and turnaround time [8]. The C.P.(Call Procedure) is used to dispatching the task according to the capability of the machines. In this model server machine is assume to make up of n heterogeneous processing elements using the cluster. Every processing element can run one task at a time and all tasks can be assigning to any node. In the proposed model subtasks communicate to each other to share the data, so execution time is reduced due to the sharing of the data. These subtasks assign to the server which dispatch the tasks to the different nodes. The scheduling algorithm is used to compute the execution cost and communication cost. So the server is assumed by a system (P,[Pij],[Si],[Ti],[Gi],[Kij]) as follows:

a) *P={Pi,… ..,Pn}*// **where Pi denotes the processing elements on cluster.**
b) *[Pij]*,**where nxn is processor topology.**
c) *Si* ,**1<=i<=n, specify the speed of processors *Pi*.**
d) *Ti,* **1<=i<=n, specify the startup cost of initiating a message on *Pi*.**
e) *Gi,* **1<=i<=n, specify the startup cost of initiating a process on *Pi*.**
f) *Kij, is the transmission rate over the link connecting two adjacent processors Pi and Pj.*

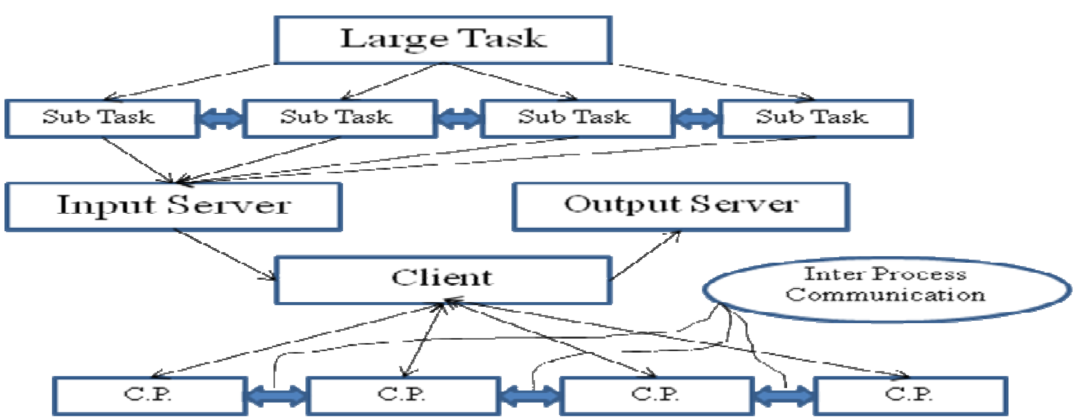

Figure 2: Proposed Dynamic Task Partitioning Model

The model comprises the existence of an I/O element associated with each processor in the system. The processing time may be executed with help of the Gantt Chart. The connectivity of the processing element can be represented using an undirected graph called the scheduler machine graph [7]. The C.P.(Call Procedure) are used to assign the task dynamically. Task can be assign to a processing element for execution while this processing element is communicating with another processing element. Program completion cost can be computed as:

$$\textbf{Total Cost} = \textbf{communication cost} + \textbf{execution cost}$$

Where:

- Execution cost=Schedule length
- Communication cost=the number of node pairs (u,v) such that (u,v)∈A and proc(u)=proc(v).

2.1 Algorithm used for the proposed model:

An optimal algorithm for scheduling interval ordered tasks on m processor. A task graph G=(V,A) and m processors, the algorithm generates a schedule f that maps each task v∈V, to a processor Pv and a starting time tv. The communication time between the processor Pi and Pj may be defined as

$$\mathbf{Comm(i,j)} = \begin{cases} \mathbf{0\ for\ \ i = 1} \\ \mathbf{1\ otherwise} \end{cases}$$

- task-ready(v,i,f):the time when all the messages from all task in N(v) have been received by processor Pi in schedule f.
- start time(v,i,f):the earliest time at which task v can start execution on processor Pi in schedule f.
- proc(v,f):the processor assign to task v in schedule f.
- start(v,f):the time in which task v begins its actual execution in schedule f.
- task(i,t,f):the task schedule on processor Pi at time t in schedule f .If there is no task schedule on processor Pi at time t in schedule f ,then task(i,t,f) returns the empty task Φ. Its assume that n_2(Φ)< n_2(v) .

2.2 Proposed Algorithm for Inter-Process Communication Amongst the Tasks:

In this algorithm the task graph generated and the edge cut gain parameter is considered to calculate the communication cost amongst the tasks [9].

$$\mathbf{gain(i,j)} = \mathbf{€.gain\ edge\ cut} + \mathbf{(1 - €)gain\ balance}$$

gain edgecut=£. newedgecut/ old edgecut

£.edgecut=old edgecut -— new edgecut

Where € is used to set the percentage of gains from edge-cut and workload balance to the total gain.

start
task(i,t,f)←Φ,for all positive integers i, where 1≤i≤m and t≥0
repeat
let v be the unmark task with the highest out-degree in v
for i=1 to m do
task-ready(v,i,f)←max((start(v,f))+comm(proc(v,f),i)+1)+gain(i,j), ∀ v∊N(v)
where gain(i,j)=€.gain edgecut+(1-€)gain balance
start time(v,i,f) ←min t, where(task(i,t,f)= Φ and t≥ task-ready(v,i,f))
end
for
f(v) ←(i, start time(v,i,f)) if
start time(v,i,f) < start time(v,j,f), 1≤j≤m, i ≠ j or
start time(v,i,f) =start time(v,j,f) and
n_2(task (i, (start-time(v,i,f)-1),f) ≤ n_2(task (j, (start-time(v,j,f)-1),f), 1≤j≤m, i ≠ j
mark task v until tasks in v are marked
end

The bigger €, the higher percentage of edge-cut gain contribute to the total gain of the communication cost.

3. CONCLUSION AND FUTURE WORK

In this paper, we proposed a new model for estimating the cost of communication amongst the various nodes at the time of the execution. Our contribution gives cut edge inter-process communication factor which is highly important factor to assign the task to the heterogeneous systems according to the processing capabilities of the processors on the network. The model can also adapt the changing hardware constraints. The researchers can improve the gain percentage for the inter process communication.

REFERENCES

[1] N. Islam and A. Prodromidis and M. S. Squillante, ‘‘Dynamic Partitioning in Different Distributed-Memory Environments’’, Proceedings of the 2nd Workshop on Job Scheduling Strategies for Parallel Processing, pages 155-170,April 1996.

[2] David J. Lilja, ‘‘Experiments with a Task Partitioning Model for Heterogeneous Computing,’’ University of Minnesota AHPCRC Preprint no. 92-142, Minneapolis, MN, December 1992.

[3] L. G. Valiant. ‘‘A bridging model for parallel computation’’. Communications of the ACM, 33(8):103-111, August 1990.

[4] B. H. H. Juurlink and H. A. G. Wijshoff. ‘‘Communication primitives for BSP Computers’’ Information Processing Letters, 58:303-310, 1996.

[5] H. EI-Rewini and H.Ali, ‘‘The Scheduling Problem with Communication’’ ,Technical Report ,University Of Nebraska at Omaha,pp 78-89,1993.

[6] D. Menasce and V. Almeida, ‘‘Cost-Performance Analysis of Heterogeneity in Supercomputer Architectures ’’, Proc. Supercomputing ’90, pp. 169-177, 1990.

[7] T.L. Adam, K.M. Chandy, and J.R. Dickson, “A Comparison of List Schedules for Parallel Processing Systems,” Comm. ACM, vol. 17, pp. 685-689, 1974.

[8] L. G. Valiant. ‘‘A bridging model for parallel computation’’. Communications of the ACM, 33(8):103-111, August 1990.

[9] H. El-Rewini,T. G. Lewis, Hesham H.Ali, ''Task Scheduling in Parallel and Distributed Systems",Prentice Hall Series in Innovative Technology,pp 48-50.1994.

[10] M. D. Ercegovac, ''Heterogeneity in Supercomputer Architectures,'' Parallel Computing, No. 7, pp.367-372, 1988.

[11] P.B. Gibbons. A more practical pram model. In Pro-ceedings of the i989 Symposium on Parallel Algorithms and Architectures, pages 158-168, Santa Fe, NM, June 1989.

[12] Y. Aumann and M. O. Rabin. ''Clock construction in fully asynchronous parallel systems and PRAM simulation". In Proc. 33rd IEEE Symp. on Foundations of Computer Science, pages 147-156, October 1992.

[13] R. M. Karp and V. Ramachandran. ,'' Parallel algorithms for shared-memory machines". In J. van Leeuwen, editor, Handbook of Theoretical Computer Science, Volume A, pages 869-941. Elsevier Science Publishers B.V., Amsterdam, The Netherlands, 1990.

BIOGRAPHY AUTHORS:

1) Dr. Rafiqul Zaman Khan:

Dr. Rafiqul Zaman Khan is presently working as a Associate Professor in the Department of Computer Science,Aligarh Muslim University,Aligarh

He is presently working as a Associate Professor in the Department of Computer Science at Aligarh Muslim University, Aligarh, India. He received his B.Sc Degree from M.J.P Rohilkhand University, Bareilly, M.Sc and M.C.A from A.M.U. and Ph.D (Computer Science) from Jamia Hamdard University.

He has 18 years of Teaching Experience of various reputed International and National Universities viz

King Fahad University of Petroleum & Minerals (KFUPM), K.S.A, Ittihad University, U.A.E, Pune University, Jamia Hamdard University and AMU, Aligarh. He worked as a Head of the Department of Computer Science at Poona College, University of Pune. He also worked as a Chairman of the Department of Computer Science, AMU, Aligarh.

His Research Interest includes Parallel & Distributed Computing, Gesture Recognition, Expert Systems and Artificial Intelligence. Presently 04 students are doing PhD under his supervision. He has published about 25 research papers in International Journals/Conferences. Names of some Journals of repute in which recently his articles have been published are International Journal of Computer Applications (ISSN: 0975-8887), U.S.A, Journal of Computer and Information Science (ISSN: 1913-8989), Canada, International Journal of Human Computer Interaction (ISSN: 2180-1347), Malaysia, and Malaysian Journal of Computer Science(ISSN: 0127-9084), Malaysia. He is the Member of Advisory Board of International Journal of Emerging Technology and Advanced Engineering (IJETAE), Editorial Board of International Journal of Advances in Engineering & Technology (IJAET), International Journal of Computer Science Engineering and Technology (IJCSET), International Journal in Foundations of Computer Science & technology (IJFCST) and Journal of Information Technology, and Organizations (JITO).

2) Javed Ali:

Javed Ali is a research scholar in the Department of Computer Science, Aligarh Muslim University, Aligarh. He born in a village Dattoly Rangher in Saharanpur District. Uttar Pradesh, India. His research interest include parallel computing in distributed systems. He did Bsc(Hons) in mathematics and MCA from Aligrah Muslim University, Aligarh. He awarded by the State Scientist Award by Indian National Congress, India.