# A Review of Various Approaches for the Coordination Among Nodes in Distributed Systems

Manjula K
Assistant Professor, Dept of CSE, DBIT,
Bangalore,India

Dr. S Meenakshi Sundaram,
Professor & Head , Dept of CSE
GSSIETW, Mysuru,India

***Abstract*— Distributed systems are the form of a network where the decisions regarding allocation of tasks and load balancing has gained importance. Extensive research surveys have been done in this field of distributed systems. An exhaustive survey related to the coordination mechanisms amongst the nodes requires to be done. Thus to cater to the necessity of the in depth research work, addressing the issue of coordination in distributed systems, this survey mainly concentrates on understanding the different techniques available to bring coordination amongst the nodes. Through this survey a deeper insight about the issue of bringing coordination amongst the nodes is addressed.**

***Keywords: Distributed systems; coordination; nodes; task allocation; load balancing,***

## I. INTRODUCTION

A distributed system is a model in which components located on networked computers communicate and coordinate their actions by passing messages. The components interact with each other in order to achieve a common goal. The availability of cheap Microprocessors (PCs, Workstations) and continuous advances in the field of Communication technology has made distributed systems gain importance [1][2][3]. Distributed system model consists of heterogeneous computers connected in an arbitrary fashion by a communication network. Computers will be having the same processing capabilities where a job may be processed from at any computer in the system from beginning towards end. Each computer can be modeled as an M/M/1 queueing system (i.e. Poisson arrivals and exponentially distributed processing times) [14] and is characterized by its average processing rate. Jobs are generated by users and they arrive at the system according to a time invariant Poisson process with average rate . The total job arrival rate must be less than the aggregate processing rate of the system, where it is assumed that the decision to distribute jobs to computers is static and will not depend on the current state of the system. Thus it becomes necessary to minimize the job's expected response time at each computer. Hence it becomes necessary to find a fair and Pareto optimal allocation.[12]

The categories of distributed systems are peer to peer, pervasive systems, cloud computing systems, grid etc. Many of the applications of distributed systems are in Telecommunication networks, network applications, realtime process control and in parallel computation. In general in any applications of distributed systems task are divided into various subtasks and these are assigned to the nodes available in the network. Jobs in a distributed system can be divided into different classes (multi-class or multi-user): (i) in multi-user type, each user jobs are grouped together; (ii) in multi-class type, all the jobs of the same size or of the same arrival rate are grouped together. The objective of the load balancing schemes can be to provide (1) a system-optimal solution where all the jobs are regarded to belong to one group (single-class) (2) an individual-optimal solution where each job optimizes its own expected response time (3) a class-optimal (user-optimal) solution where the jobs are classified into a finite number of classes (users) and each user tries to optimize the expected response time of his own jobs.[11]

The various types of load balancing policies: static policies and dynamic policies. Static policies make their decisions on statistical information collected about the system. The current state of the system will not be taken into consideration. Dynamic policies will be based on the decisions of the current state of the system.The computers exchange this information periodically and will try to balance the load by moving some jobs from heavily loaded nodes to lightly loaded nodes. Even if the runtime complexity is higher, dynamic policies will always lead to better performance when compared to static policies. The efficiency of the distributed systems will be evaluated based on the fair allocation of jobs to the nodes in the network. Fair allocation can be done using various scheduling algorithms available for distributed systems. Many related survey is made on different types of distibuted systems like survey on grid computing systems, survey on Cloud computing systems but insight into a particular issue is taken and study on it is made in this survey.

## II COORIDNATION MECHANISMS IN DISTRIBUTED SYSTEMS

As distributed systems might comprise of heterogeneous nodes to provide varying applications it becomes a complex task to make nodes in the heterogeneous network to bring coordination among the nodes. Thus this issue of noncoperating nodes not willing to communicate with the other nodes is addressed in many of the research work. By employing various stratergies noncoperative nodes are made to cooperate and coordiante to improve the overall performance of the network and identifying set of nodes willing to share their resources for executing the tasks.

### *A: Game Theory Approach*

A game theoritic framework for obtaining user-optimal load balancing scheme in heterogeneous system is presented [4] where the static load balancing among the heterogeneous nodes in distributed systems is treated as a noncoperative game among users. Principle of Nash-Equilibrium is achieved to balance the load among the noncoperative nodes. There are different approaches based on gametheory model:

1) Cooperative approach

In the cooperative approach several decision makers (computers, jobs) will be made to cooperate in taking decisions to allow the nodes operate optimally. All the decision makers will be made to communicate to make joint agreements for operating which can be modeled using a framework using game theory[5] .The static load-balancing problem in single class job distributed systems as is modeled as a cooperative game among computers. The computers comprising the distributed system are modeled as M/M/1 queuing systems. It is shown that the Nash bargaining solution (NBS) provides an optimal solution (operation point) for the distributed system and it is also a fair solution. [6] In a cooperative game, players will have complete freedom of preplay communication to make joint agreements . A cooperative game consists of M players, A non-empty, closed and convex set which is a set of stratergies of M players, and for each player I, i=1,2,….M, an initial agreement point is made.[6]

2) Non Cooperative approach

In the non cooperative approach several decision makers (computers, jobs) will not be allowed to cooperate in taking decisions. Each decision maker will try to optimize its own response time to reach Nash equilibrium [8] .The decision maker will not get any benefit by changing the decision and if the number of decision makers is not finite Nash equilibrium reduces to the Wardrop equilibrium [7] . The job allocation problem will be considered as a non-cooperative game among the users, where each user j (j = 1, . . . , m) will find the workload that is assigned to computer i such that the expected price of the jobs is minimized. The vector $\beta_j = [\beta_{j1}, \beta_{j2}, \ldots, \beta_{jn}]$ is called the job allocation strategy of user j (j = 1, . . . , m) and the vector β = [β 1 , β2 , . . . , βm] is called the strategy profile of the job allocation game. The strategy of user j depends on the job allocation strategies of the other users. The assumptions for the existence of a feasible strategy profile are as follows: (i) Positivity: β j i ≥ 0, i = 1, . . . , n, j = 1, . . . , m; (ii) Conservation: Pn i=1 β j i = φ j , j = 1, . . . , m; (iii) Stability: Pm j=1 β j i < µi , i = 1, . . . , n; A Non-cooperative job allocation game consists of a set of players, a set of strategies, and preferences over the set of strategy profiles:[11]

### *B: Graph Theory Approach*

In an augmented task graph is used which is represented as a matrix A(n+m)x(n+m) from $R_{ax,}$, $P_n$, $C_{nxn}$, and $U_{nxm}$ as follows. Each vertex of ATG corresponds to either a process or a resource[8].Vertices 1 to n of ATG correspond to processes p1 to pn, respectively and vertices n+l to n+m of ATG correspond to resources r to rr, respectively.

In [9] tasks are modelled as task interaction graph, where the vertices of the graph correspond to the tasks and the edges correspond to the intertask communications, using which sum of the total execution and communication costs are minimized in order to optimize system utilization.

## III SUMMARY

Amongst the two approaches which are used graph theory based tools are mainly used for static allocation and it is limited to a small network whereas game theory based approaches can be used for various scenarious of distributed systems to achieve coordination amongst the jobs and contribute for task allocation.

## IV CONCLUSION

Distributed systems are currently developing rapidly and new arena of distributed computing is getting evolved with emerging big data which needs a high efficiency interoperability and coordination amongst the cluster of nodes. Thus it becomes necessary to consider these emerging trends for task allocation and load balancing.

## REFERENCES

[1] Yichuan jiang,Zhichuan Huang,"The Rich Get Richer: Preferential Attachment in the Task Allocation of Cooperative Networked Multiagent Systems with Resource Caching", IEEE Transactions on Systems, Man and Cybernetics-Part A:Systems and Humans, 42(5):1040-1052,2012

[2] Yichuan Jiang , Jing Hu, Donghui Lin, "Decision Making of Networked Multiagent Systems for Interaction Structures," IEEE Transactions on Systems, Man and Cybernatics A: Systems and Humans, 41(6): 1107-1121,2011

[3] Dayong Ye, Minjie Zhang, Danny Sutanto," Self-Adaptation Based Dynamic Coalition Formation in A Distributed Agent Network: A Mechanism and A Brief Survey", IEEE Transactions on Parallel and Distributed Systems, 24(5): 1042-1051, 2013

[4] Daniel Grosu, A.T Chronopoulous, "Noncoperative Load Balancing in Distributed Systems,"Journal of Parallel and Distributed Computing,65(9):1022-1034,2005

[5] Kim C, Kameda H. Optimal static load balancing of multiclass jobs in a distibuted computer system. Proceedings of the 10th International Conference on Distributed Computing System

[6] Cooperative load balancing in distributed systems D. Grosu[1], Chronopolous[2]

[7] Roughgarden T, Tardos E. How bad is selfish routing? Proceedings of the 41st IEEE Symposium on Foundations of Computer Science, Redondo Beach, CA, November 2000; 93-102

[8] Bharadwaj v, Ghose D , Mani V, Robertazzi TG Scheduling Divisible Loads in Parallel and Distributed Systems. IEEE Computer Society Press: Loas Alamitos, CA, U.S.A., 1996

[9] Task aaignement in heterogeneous computing systems Bora U cara ,Cevedet Ayakanat [a] ,Kamer Kaya[a] ,Murat Ikhinch[b]

[10] Static task allocation for heterogeneous Distributed Computing Systems S. Selvakumar C. Siva Ram Murthy Department of Computer Science and Engineering Indian Institute of Technology Madras 600 036, India

[11] Game theory based job allocation/load balancing in distributed systems with applications to grid computing Satish Penmatsa,M.S.

[12] A Truthful Mechanism for Fair Load Balancing in Distributed Systems Daniel Grosu and Anthony T. Chronopoulos Department of Computer Science

[13] D. Gross and C. M. Harris, Fundamentals of Queueing Theory, Wiley-Interscience, New York, NY, 1998.

[14] N. Nisan, S. London, O. Regev, and N. Camiel, " Globally Distributed Computation over Internet – The POPCORN Project", Proc. of the 18th IEEE International Conference on Distributed Computing Systems, pp. 592-601, May 1998