# Scheduling in Heterogeneous Distributed Systems using Game Theory

Santosh Aditham, *Member, IEEE*

***Abstract*—Task scheduling in Heterogeneous Distributed Systems (HDS) is an NP-complete problem. However, a lot of research work is being done in this area from the past many years, to come up with algorithms and techniques that assign tasks to processors in the best possible way. The problem of scheduling in HDS can be modeled as assigning tasks that are different from each other, to processors of the HDS so that no task can benefit by changing its processor assignment provided all other assignments remain unchanged. Such a problem can be identified to the game theory problem where players in an n-player game choose their strategy such that Nash Equilibrium is achieved. The idea in this paper comes in two steps. First step is to represent the tasks in HDS using functional descriptions of task graphs and dynamically updatable execution profiles. Second step is to tackle the scheduling problem by using 'First Price Sealed-bid Auction' in an aggressive way with a sliding window. Experimental results to show the effectiveness of the proposed scheme will be provided.**

***Index Terms*— Distributed system, Grid, scheduling, tasks, functional description, execution profile, cooperative game, Nash Equilibrium**

## I. INTRODUCTION

THE technology world, as we see it today, is fully interconnected and the need of a central power house seems inevitable. A Heterogeneous Distributed System (HDS) is the idea of having geographically separated computers with diverse capabilities connected together as a single super computer that is capable of solving even the most complex jobs. When a large, complex job is submitted to a HDS, it is broken down into small independent tasks. Inherent to such an idea lays the obvious importance of task representation and management.

A task graph is a directed acyclic graph with nodes as tasks and arcs as data dependencies. Representing tasks using task graphs is a common practice but the ever increasing complexity and scale of the jobs makes the approach cumbersome. A functional description of a task graph is nothing but a k-tuple that contains all possible paths in a given task graph. Identifying parallel executable tasks is possible by parsing through the functional description. Also, dynamically modifying the functional description is easy. At a given point, the set of enabled or active tasks in a functional description constitute to the 'root tasks'. These root tasks are independent to each other and can be executed parallel using the available machines of HDS. An Execution Profile is a normalized functional statement, which has a possible execution path including at least one initial task. It can be dynamically modified as the execution progresses [2,3]. Steps to obtain an Execution profile from task graph will be discussed in this paper. In order to schedule the current set of root tasks from the execution profile and execute them dynamically, we need a lot of information like precedence conditions among tasks, availability of machines, priority of tasks etc. This paper supports the idea of using execution profiles to represent the jobs in a HDS.

This leads to the next step, solving the task mapping problem which is a known NP-complete problem. Task mapping is the combination of task assignment and task scheduling of tasks [1]. Task assignment is the process of allocating the most suitable machine, out of all the available machines, to execute a given independent task. Because of the infeasibility that a HDS can be fully connected and the limited information regarding the HDS to a machine, task assignment becomes probabilistic. In this work, similarities of task assignment problem to game theory problems will be explored. In Game Theory, a game is the interaction among rational, mutually aware players, where the decisions of some players impact the payoffs of others [4]. To make the game dynamic, the decision process should evolve with time and players should have preferences and access to more information. Dynamic task allocation can be compared to such a dynamic game where the tasks behave as players. Task scheduling is the process of planning the execution of all tasks so that we achieve maximum performance for the entire HDS. Task scheduling can be centralized or decentralized or hybrid, and static or dynamic.

---

Assumptions in this work are: all computers in the grid are always in perfectly working condition and any computer will completely execute a task when it is assigned with one. Future work will focus on fault-tolerance. Terms like 'computer', 'machine' and 'processor' are used interchangeably.

This paper supports the idea of hybrid, dynamic scheduling. The main inspiration for representing tasks in HDS using execution profiles comes from [2] and for using game theory to solve the task mapping problem comes from [1]. Representing the job in a HDS using dynamically updatable execution profile and scheduling the tasks dynamically using first price sealed bid auction is the central idea of this paper. Root tasks are modeled as buyers competing to buy active machines in the HDS from the seller. The seller would be a central static scheduler. Execution time of a root task on a machine is the private value in the bid. Buyers follow either a conservative or an aggressive strategy. The conservative approach is static. In the aggressive approach, every task maintains a list of active machines and selects the machines that can provide earlier completion time compared to the static assignment. Nash Equilibrium is then calculated to solve the game.

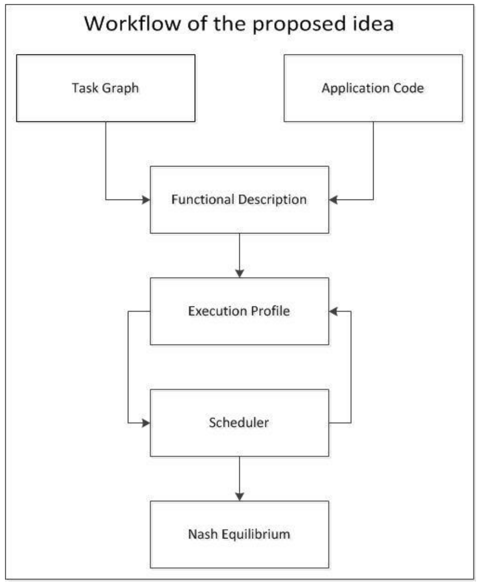


Fig 1. Workflow of the proposed idea

This term paper proposal is organized as follows: Section II talks about related work in the fields of scheduling in HDS, game theory and task graph representations. Section III gives the outline of how the final report would be written and organized. Section IV will give the conclusion and future work for phase2 submission of this term paper.

## II. Related Work

As mentioned before, a lot of research work is being done in the field of scheduling in HDS. Game theory is being vastly explored as an efficient way to handle the scheduling problems in HDS from the 1980s and 90s. More recent works emphasize on modeling the HDS scheduling problem as a cooperative or non-cooperative game, attaining Nash Equilibrium, conducting auctions and bidding for machines. Load balancing is a terminology used in congruence to task allocation and it emphasizes on resource utilization.

In [6], the author proposes a task allocation approach which distributes tasks to processors based on task entropies such that all resources are consumed. The values of task entropies relate to the probabilistic distribution of the tasks and the structure of task allocation procedure. The author proposed few gain functions which would measure the benefits of choosing a machine to execute as task. These gain functions work based on bids from processors.

In [7], the authors propose semi-static, hybrid game-theoretic-based solution to the grid load balancing problem. This work models the problem as a non-cooperative game with an objective to reach the Nash Equilibrium. The focus is to minimize the average task completion time. They analyzed the effects of system load, size, communication delay and service time using simulations. In another work [], the authors tackle the exact same problem as above but they concentrated on having multiple communication channels between machines.

The concept of Selfish Grids proposed in [8] is another interesting take on the same HDS scheduling problem. The authors propose a hierarchical semi-selfish game model with intrasite job execution game, intrasite bidding game and intersite bidding game. They claim that selfish behavior of machines is not a bad sign to the grid if they all use the same optimal mixed strategy. They also claim that the Nash Equilibrium method is only as good as a totally uncoordinated way of scheduling jobs.

The other type of scheduling in HDS is called multi-agent scheduling. Project scheduling is a cooperative game theory approach where rewards and penalties by the customer are shared among multiple agents. In [9], the authors propose some strategies and decision problems which model the scheduling problem in HDS as a game theoretic problem. They show how to characterize the best Nash Equilibrium strategies using multi-agent project scheduling.

Coalitional game is an approach suggested in [10] for divisible load scheduling in a distributed system interconnected with a bus network. This paper proposes a cooperative game theory among organizations that own machines and job owners. The aim here is to minimize the total execution time of the scheduling problem, referred as BUS-LINEAR problem in this paper.

The other set of game theoretic solutions to the scheduling problem in HDS involve auctions and bidding. Auction is one way of market mechanism in economic game theory which is played by a set of rules and exchange of resources is done by having bids. It is a less complicated method in reference to large systems like HDS because auctions are decentralized and players do not need a lot of information. Also, auctions can support requests involving bundles/groups of resources instead of separating the resource requirements. Various mechanisms of auctions are: first-price, second-prize, English and Dutch auctions [5].

In [11], a repeated auction based protocol is presented for sharing grid resources. The authors claim that bidding improves long term profits of grid users. The paper introduces various *fees* in order to hold the auctions. The role of a manager is required here, to conduct the auction as a cooperative game. Here the users play the auction and bid for processors. The users who play the auction employ certain learning algorithms to learn strategies that help them place a winning bid. Another auction based algorithm is suggested in [12] that does resource allocation while trying to keep the cost and time optimization at minimum. This paper also introduces certain other ideas such as having bag of tasks and associating users to brokers. Similarly, in [13] BarSAA algorithm is proposed where prices of resources are negotiated iteratively during an auction and adjusted dynamically to reach equilibrium. It is called a socially efficient allocation mechanism to achieve competitive equilibrium in the system. Even though [14] is about cloud computing, they introduce an interesting concept of predicting resource price by the user, based on history of previous bids.

## III. Outline for Final Report

### A. *Why I want to work on this problem?*

Even though this work concentrates on HDS, it can be a good learning towards dealing with other scheduling problem domains like multi-core processors and cloud computing. Since we know that the scheduling problem is NP-complete, there is no best solution to this problem and hence makes it an interesting area to work on. Also, finding solutions to the scheduling problem using game theory is an area that is relatively less explored.

### B. *Basic version of my algorithms*

<u>Algo 1:</u>

```
Ri = set of all active tasks at level i
While (Ri ≠ ∅)
{
        Mark each task Ti ∈ Ri that is not scheduled, as scheduled
        Verify conditions required for each Ti
        If (conditions = true) then
                Add Ti to set of root tasks, S'
                Call Algo 2
                Remove any completed tasks C' from set of root tasks, S'
        Update Execution Profile
}
```

<u>Algo 2:</u>

```
For each si ∈ S'
{
        Conservative[si] = Static-scheduler(si ∈ S')
        For mi ∈ set of all active machines, M'
        {
                If(window(si) within idle(mi)) then
                        Add mi to list Candidates[si]
        }
}
Aggressive[si] = BestCandidate(Candidates[si])
P' = Calculate Payoffs(S', Conservative[], Aggressive[])
Dynamic-scheduler(si ∈ S') = NashEquilibrium(S',P')
C' = Schedule (Dynamic-scheduler(si ∈ S'))
```

### C. *How will my final report look like?*

The final report will explain in detail about:

1. The framework for solving the scheduling problem in HDS.
2. How a given large and complex job can be divided into small independent tasks.
3. How the set of independent tasks can be represented in a dynamically updatable Execution Profile.
4. Given an execution profile, how to efficiently allocate the tasks that are ready to machines that are available.
5. How to find the Nash Equilibrium in task allocation.
6. How to schedule the execution of tasks in order to achieve maximum throughput.
7. Coding and implementation of all the above mentioned ideas.
8. Experimental results with graphs and comparisons.
9. Conclusion and future work.

The final report will be organized as:
*abstract*
Section I – Introduction
Section II – Related Work
Section III – Model and Algorithms
Section IV – Experiments and Results
Section V – Conclusions
Bibliography

## IV. CONCLUSION

As of now, this proposal only talks about the basic ideas being considered to implement a fully functional and effective way of scheduling in HDS, using game theory concepts and functional descriptions of task graphs. The proposal introduces the problem and the idea behind the solution. The proposal also gives a detailed walk-through about some related work done in the same area. The proposal touches a little bit on how to build a framework for solving the scheduling problem of the HDS. Future work will be talking more about:

1. Final versions of the algorithm(s) needed;
2. Coding and implementing of such algorithms;
3. Analyzing the experimental results and
4. Conclusions.

## REFERENCES

[1] Vasanth Kumar Ramesh, *A game theoretic framework for dynamic task scheduling in distributed heterogeneous computing systems* (Master's Thesis). Tampa, 2005.
[2] Santosh Aditham, *Task Management driven Simulation of Grid Applications using Stochastic Petrinets* (Master's Thesis). Lubbock, 2010.
[3] Noe Lopez-Benitez, Per Anderson, *Dynamic Structures for the Management of Complex Applications in a Heterogeneous Computing System*. 2009 International Conference on Grid Computing & Applications, pp 80-85. Las Vegas, 2008.
[4] Shor, Mikhael, "*Game.*" Dictionary of Game Theory Terms, Game Theory .net, < *http://www.gametheory.net/dictionary/Game.html* > Web accessed: 10/08/2012.
[5] Shor, Mikhael, "*Auction.*" Dictionary of Game Theory Terms, Game Theory.net, <*http://www.gametheory.net/dictionary/Auctions/Auction.html*> Web accessed: 10/08/2012.
[6] Sverrir Olafsson, *A General Model for Task Distribution on an Open Heterogeneous Processor System*. IEEE Transactions on Systems, Man and Cybernetics, vol.25, no.1, pp. 43-58, Jan 1995.
[7] Riky Subrata, Albert Y. Zomaya, Bjorn Landfeldt, *Game-Theoretic Approach for Load Balancing in Computational Grids*. IEEE Transactions on Parallel and Distributed Systems, vol. 19, no. 1, pp. 66-76, Jan 2008.
[8] Yu-Kwong Kwok, Kai Hwang, ShanShan Song, *Selfsih Grids: Game-Theoretic Modeling and NAS/PSA Benchmark Evaluation*. IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 5, pp. 621-636, May 2007.
[9] Cyril Briand, Jean-Charles Billaut, *Cooperative project scheduling with controllable processing times: a game theory framework*. IEEE ETFA, 2011.
[10] Thomas E. Carroll, Daniel Grosu, *Divisible Load Scheduling: An Approach using Coalitional Games*. ISPDC, 2007.
[11] Amir Danak, Shie Mannor, *Efficient Bidding in Dynamic Grid Markets*. IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 9, pp 1483-1496, Sep 2011.
[12] Zhao Lan, Zheng Zhao-xia, *A Deadline and Cost Optimization Algorithm for Scheduling Tasks in Grids and Nash Equilibrium in Auction-based Systems*. CCDC, 2008.
[13] Han Zhao, Xiaolin Li, *Efficient Grid Task-Bundle Allocation using Bargaining based Self-adaptive Auction*. 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009.
[14] Fei Teng, Frederic Magoules, *Resource Pricing and Equilibrium Allocation Policy in Cloud Computing*. 10th IEEE International Conference on Computer and Information Technology, pp 195-202. 2010.

## BIBLIOGRAPHY


[1] Vasanth Kumar Ramesh, *A game theoretic framework for dynamic task scheduling in distributed heterogeneous computing systems* (Master's Thesis). Tampa, 2005.

[2] Santosh Aditham, *Task Management driven Simulation of Grid Applications using Stochastic Petrinets* (Master's Thesis). Lubbock, 2010.

[3] Noe Lopez-Benitez, Per Anderson, *Dynamic Structures for the Management of Complex Applications in a Heterogeneous Computing System*. 2009 International Conference on Grid Computing & Applications, pp 80-85. Las Vegas, 2008.

[4] Shor, Mikhael, "*Game*." Dictionary of Game Theory Terms, Game Theory .net, < *http://www.gametheory.net/dictionary/Game.html* > Web accessed: 10/08/2012.

[5] Shor, Mikhael, "*Auction*." Dictionary of Game Theory Terms, Game Theory.net, <*http://www.gametheory.net/dictionary/Auctions/Auction.html*> Web accessed: 10/08/2012.

[6] Sverrir Olafsson, *A General Model for Task Distribution on an Open Heterogeneous Processor System*. IEEE Transactions on Systems, Man and Cybernetics, vol.25, no.1, pp. 43-58, Jan 1995.

[7] Riky Subrata, Albert Y. Zomaya, Bjorn Landfeldt, *Game-Theoretic Approach for Load Balancing in Computational Grids*. IEEE Transactions on Parallel and Distributed Systems, vol. 19, no. 1, pp. 66-76, Jan 2008.

[8] Yu-Kwong Kwok, Kai Hwang, ShanShan Song, *Selfsih Grids: Game-Theoretic Modeling and NAS/PSA Benchmark Evaluation*. IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 5, pp. 621-636, May 2007.

[9] Cyril Briand, Jean-Charles Billaut, *Cooperative project scheduling with controllable processing times: a game theory framework*. IEEE ETFA, 2011.

[10] Thomas E. Carroll, Daniel Grosu, *Divisible Load Scheduling: An Approach using Coalitional Games*. ISPDC, 2007.

[11] Amir Danak, Shie Mannor, *Efficient Bidding in Dynamic Grid Markets*. IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 9, pp 1483-1496, Sep 2011.

[12] Zhao Lan, Zheng Zhao-xia, *A Deadline and Cost Optimization Algorithm for Scheduling Tasks in Grids and Nash Equilibrium in Auction-based Systems*. CCDC, 2008.

[13] Han Zhao, Xiaolin Li, *Efficient Grid Task-Bundle Allocation using Bargaining based Self-adaptive Auction*. 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009.

[14] Fei Teng, Frederic Magoules, *Resource Pricing and Equilibrium Allocation Policy in Cloud Computing*. 10th IEEE International Conference on Computer and Information Technology, pp 195-202. 2010.

[15] Han Zhao, Xinxin Liu, Xiaolin Li, *Hypergraph-based Task-Bundle Scheduling Towards Efficiency and Fairness in Heterogeneous Distributed Systems*. 2010.

[16] Satish Penmatsa, Anthony T. Chronopoulos, *Job Allocation Schemes in Computational Grids Based on Cost Optimization*. Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, 2005.

[17] Joanna Kołodziej, Fatos Xhafa, *Modelling of User Requirements and Behaviors in Computational Grids*. 2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp 548-553. 2010.

[18] Massimo Orazio Spata, *A Nash-Equilibrium based Algorithm for Scheduling Jobs on a Grid Cluster*.

[19] Setareh Nouri, Saeed Parsa, *A Non-cooperative Approach For Load Balancing In Heterogeneous Distributed Computing Platform*. 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology, pp 756-761. 2009.

[20] Samee Ullah Khan, Ishfaq Ahmad, *Non-cooperative, Semi-cooperative, and Cooperative Games-based Grid Resource Allocation*.2006.

[21] Joel Wilkins, Ishfaq Ahmad, Hafiz Fahad Sheikh, Shujaat Faheem Khan, Saeed Rajput, *Optimizing Performance and Energy in Computational Grids using Non- Cooperative Game Theory*.2010.

[22] Preetam Ghosh, Kalyan Basu, Sajal K. Das, *A Game Theory-Based Pricing Strategy to Support Single/Multiclass Job Allocation Schemes for Bandwidth-Constrained Distributed Computing Systems*. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, vol. 18, no. 3, pp 289-306. March 2007.

[23] Peng Xiao, Zhigang Hu, *A Novel QoS-based Co-allocation Model in Computational Grid*. IEEE "GLOBECOM" 2008.

[24] Riky Subrata, Albert Y. Zomaya, Bjorn Landfeldt, *A Cooperative Game Framework for QoS Guided Job Allocation Schemes in Grids*. IEEE TRANSACTIONS ON COMPUTERS, vol. 57, no. 10, pp 1413-1422. OCTOBER 2008.

[25] Sangho Yi, Emmanuel Jeannot, Derrick Kondo, David P. Anderson, *Towards Real-Time, Volunteer Distributed Computing*. 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp 154-163.2011.

[26] Guiyi Wei, Athanasios V. Vasilakos, Naixue Xiong, *Scheduling Parallel Cloud Computing Services: An Evolutional Game*. The 1st International Conference on Information Science and Engineering (ICISE2009), pp 376-379. 2009.

[27] Ishfaq Ahmad, Sanjay Ranka, Samee Ullah Khan, *Using Game Theory for Scheduling Tasks on Multi-Core Processors for Simultaneous Optimization of Performance and Energy*. 2008.

[28] Joanna Kołodziej, Fatos Xhafa, Marcin Bogdanski, *A Stackelberg Game for Modelling Asymmetric Users' Behavior in Grid Scheduling*. 12th International Conference on Computer Modelling and Simulation, pp 497-502. 2010.

[29] Fatos Xhafa, Joanna Kołodziej, *Game-theoretic, Market and Meta-Heuristics Approaches for Modelling Scheduling and Resource Allocation in Grid Systems*. 2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp 235-242. 2010.


**Santosh Aditham** is a PhD student under Dr.Ranga, in the department of Computer Science and Engineering at University of South Florida. His major research areas are Computer Architecture and Distributed Systems. He is a member of the IEEE since 2012. He currently serves as the Vice President of the IEEE CS student chapter at USF.