Mechanism Design Approach For Load Balancing In Distributed Systems

Abhishek Kumar, Mayank Kumar Agrawal, Vaibhav Pratap Singh

Advisor: Prof. Gopalan Sajith

Dept. of CSE, Indian Institute of Technology Guwahati, Guwahati

Introduction

Load Balancing

Computers are connected to a Scheduler which assigns load to them. Now, given large number of jobs, the assignment of jobs to different computers (with different computation powers) should be done strategically, so as to reduce the overall completion time and optimize the load balancing too.

Problem Statement

Here we investigate the problem of designing protocols for resource allocation involving selfish agents. Using mechanism design theory, we design a truthful mechanism for solving the static load balancing problem in heterogeneous distributed systems. We prove that using the optimal allocation algorithm the output function admits a truthful payment scheme satisfying voluntary participation. We derive a protocol that implements our mechanism.

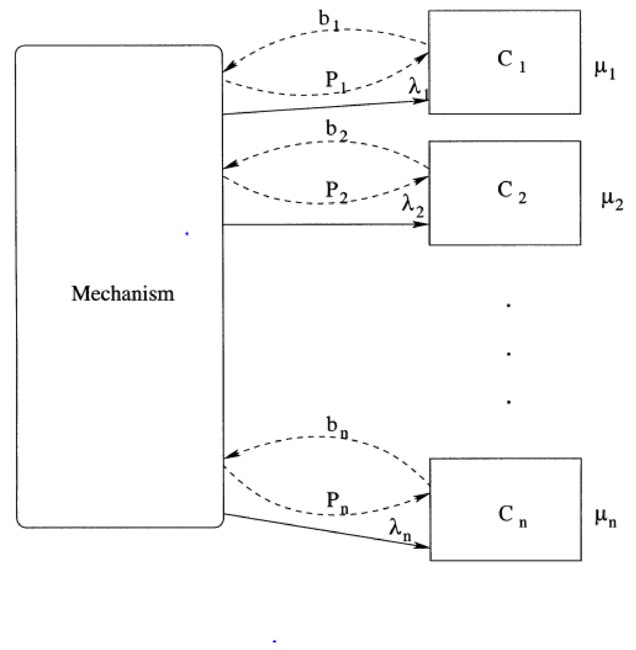

Figure 1: Load Balancing Mechanism structure

Mechanism Design

Mechanism design is a field in economics and game theory that takes an engineering approach in designing economic mechanisms or incentives, toward desired objectives, in strategic settings, where players act rationally. In a mechanism design problem the goal function is given while the mechanism is unknown.

Problem Representation

There are n PCs as agents and a global scheduler on which mechanism runs.

- True Value of Agent i: t_i
- Bid Value of Agent i: b_i
- $b = (b_1, b_2, \ldots, b_n)$
- $b_{-i} = b - b_i$
- Load to Agent i: $\lambda_i(\mathbf{b})$
- Payment to Agent i: $P_i(\mathbf{b})$
- $\lambda(\mathbf{b}) = (\lambda_1(\mathbf{b}), \ldots, \lambda_n(\mathbf{b}))$
- $P(\mathbf{b}) = (P_1(\mathbf{b}), \ldots, P_n(\mathbf{b}))$
- $cost_i(t_i, \mathbf{b}) = t_i\lambda_i(\mathbf{b})$
- $utility_i(t_i, \mathbf{b}) = P_i(\mathbf{b}) - cost_i(t_i, \mathbf{b})$

Definitions

Def₁ : Truthful Mechanism

Utility should be maximum when $\mathbf{b_i = t_i}$.

Conditions :

(i) First order derivative should be 0 at $b_i = t_i$

$utility_i' = P_i'(\mathbf{b}) - t_i\lambda_i'(\mathbf{b}) = 0$

Integrating by parts gives

$P_i(\mathbf{b}) = P_i(0) + b_i\lambda_i(\mathbf{b}) - \int_0^{b_i}\lambda_i((b_{-i}, u))du$

(ii) $utility_i'' \leq 0$, at $b_i = t_i$

$utility_i'' = P_i''(\mathbf{b}) - t_i\lambda_i''(\mathbf{b}) \implies \lambda_i'(t_i, b_{-i}) \leq 0$

Def₂ : Truthful Payment Mechanism

We say than an output function admits a truthful payment scheme if there exists a payment function P such that the mechanism is truthful. In Def_1 we have shown that if λ_i is a decreasing function with payment as $P_i(\mathbf{b}) = P_i(0) + b_i\lambda_i(\mathbf{b}) - \int_0^{b_i}\lambda_i((b_{-i}, u))du$ then our mechanism is truthful.

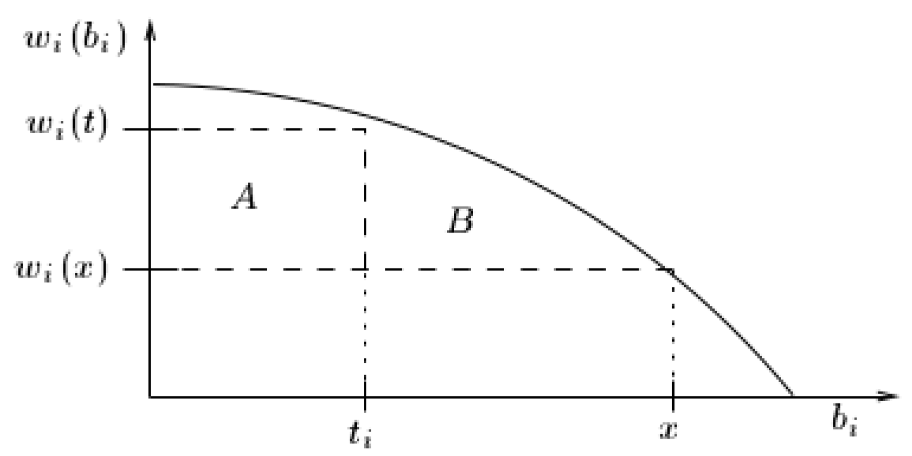

Figure 2: This picture shows why agent never gains by overbidding.

Def₃ : Voluntary Participation Mechanism

$utility_i(t_i, (b_{-i}, t_i)) \geq 0$ so that agent participates voluntarily. We need to set $P_i(0)$ to a constant greater than $\int_0^{b_i}\lambda_i((b_{-i}, u))du$

A decreasing output function admits a truthful payment scheme satisfying voluntary participation if and only if $\int_0^{\infty}\lambda_i(b_{-i}, u)du \leq \infty$ for all i. Thus we can make the payments to be $P_i(\mathbf{b}) = b_i\lambda_i(\mathbf{b}) + \int_{b_i}^{\infty}\lambda_i((b_{-i}, u))du$

Def₄ : Feasible Allocation

Goal : min $D(\lambda)$

Expected response time for load at agent i is $1/(\mu_i - \lambda_i)$, where μ_i is average processing rate of agent i and Φ be the total job arrival rate of the system.

Overall average Expected Response Time

$D(\lambda) = -(n/\Phi) + (1/\Phi)\sum_{i=1}^{n}(\mu_i)/(\mu_i - \lambda_i)$

To minimize $D(\lambda)$ we need to minimize $\sum_{i=1}^{n}(\mu_i)/(\mu_i - \lambda_i)$ under constraints:

1. Positivity: $\lambda_i \geq 0$, i= 1,...,n
2. Conservation: $\sum_{i=1}^{n}\lambda_i = \Phi$
3. Stability: $\lambda_i < \mu_i$, i = 1,...,n

Feasible region of above problem is convex, hence KKT-conditions are sufficient conditions for Global Minima. Thus we can find the optimal allocation using Lagrange Multiplier Theorm.

Algorithm

Input: Bid value $b_1, \ldots, b_n$ and total arrival rate Φ

Output: Load allocation $\lambda_1, \ldots, \lambda_n$

Desired Outcome: $D(\lambda)$ is minimized at $\lambda_i = (1/b_i) - (1/\sqrt{b_i}) * ((\sum_{i=1}^{n} 1/b_i) - \Phi) / (\sum_{i=1}^{n}\sqrt{1/b_i})$//

Psuedocode:

1. Let $b_1 \leq \ldots \leq b_n$ are sorted in increasing order.
2. prefsum1[n] = contains prefix sum of $1/b_i$
3. prefsum2[n] = contains prefix sum of $\sqrt{1/b_i}$
4. let left $\leftarrow$ 1, right $\leftarrow$ n
5. let c $\leftarrow$ 0
6. while(left $\leq$ right)
   (a) mid $\leftarrow$ (left + right)/2
   (b) c $\leftarrow$ (prefsum1[m]- Φ)/prefsum2[m]
   (c) if($\sqrt{1/b_i} \leq$ c) then right $\leftarrow$ mid - 1
   (d) else then left $\leftarrow$ mid + 1
7. let m $\leftarrow$ right - 1
8. for each i satisfying $1 \leq i \leq m$
   (a) let c $\leftarrow$ (prefsum1[mid]- Φ)/prefsum2[mid]
   (b) let $\lambda_i \leftarrow (1/b_i) - c\sqrt{1/b_i}$
9. for each i satisfying $m+1 \leq i \leq n$
   (a) let $\lambda_i \leftarrow 0$

Future Work

Further, we will introduce the cost of data communication between the computers along with caching and memory access costs. Also the effect of altruistic behaviour of the agents will be studied.

References

[1] Aaron Archer and va Tardos. Truthful mechanisms for one-parameter agents. 2001.

[2] Daniel Grosu and Anthony T. Chronopoulos. Algorithmic mechanism design for load balancing in distributed systems. 2004.

[3] Y Narhari. In *Game Theory And Mechanism Design*.

[4] Noam Nisan. Algorithmic mechanism design. 2001.