Several Approaches to the Assignment Problem in Distributed Systems

Guillermo Morales Luna
CINVESTAV-IPN

(joint work with Mario Farías Elinos)

We present some approaches to deal with the assignment problem in distributed systems.

Suppose given a set of n tasks and m processors. For each task $j \leq n$ there are specified the following numbers:

Duration. $\forall j:\ T_j$ is the total time in which task j should be completed.

Number of instructions. $\forall i \leq m : I_{ij} \in \mathbb{Z}^+$ is the number of instructions on processor i necessary to complete task j.

Utilization. $\forall i \leq m : u_{ij} \in \mathbb{R}$ is the *utilization* of processor i by the task j in case it is undertaken.

An *assignment* is a relation of the set of processors into the set of tasks. It is not a function since the tasks may be replicated.

Each assignment is determined by a matrix $V \in \{0,1\}^{m\times n}$, namely

$$\forall i,j:\ v_{ij} = \begin{cases} 1 & \text{if task } j\text{is assigned to processor } i, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

The assignment problem is:

$$\begin{aligned} \text{Minimize} \quad & f(V) = a\sum_{i=1}^{m}\left(\sum_{j=1}^{n} u_j v_{ij}\right)^2 + b\sum_{j=1}^{n}\left(\sum_{i=1}^{m} v_{ij} - r\right)^2 \\ \text{subject to} \quad & V \in \{0,1\}^{m\times n} \end{aligned} \tag{2}$$

The search space has size $2^{m\cdot n}$.

We present a comparison of methods, including simulated annealing and genetic algorithms, and we illustrate how common sense approaches are almost effective for this problem.