A Decentralized Task Scheduling Algorithm & its Performance Modeling for Computer Networks

Sagar Bijwe (07305023) **Pulkit Goyal (07305009)**

Abstract

We consider the problem of mapping tasks to processor nodes at run-time (dynamic approach) in multicomputer systems. This is against static allocation where work load allocation decisions are taken before run time and tasks are assigned to individual processors of the system. In a decentralized approach, the scheduling control is distributed amongst all the nodes in the system. The scheduling strategies in such a system should incur less overhead and identify suitable remote sites for migrating extra work.

The decentralized algorithm proposed in the paper [1], is a combination of neighborhood averaging and bidding approach and it is intended for multicomputers connected by a store and forward communication network.

The factors that characterize the performance of the system are:

- Topology used
- Stability (oscillation free or not)
- Task Trashing Level (task thrashing occurs when node spends more time in task migration than task execution)

We intend to study the performance of the system under different topologies like mesh, hypercube and Chordal Ring, and also comment on the thrashing and stability of the system.

For performance evaluation, the cost of migration and scheduling overhead is also taken into account. For comparison, we select random scheduling strategy which also makes use of nearest neighbor load balancing.

System Parameters

- Load per node.
- Number of neighbors for each node.

Performance Evaluation Metrics

- Response time of the task which can be split into two phases:
    1. Task's settling time which consists of scheduling and communication times, and waiting times in scheduling and communication queues at various nodes
    2. Waiting time in the execution queue (local queue) once node is chosen
- Throughput of the system.
- Total time spent in migration and execution by a node.
- Probability of task being scheduled locally.

Assumptions

- The task transferred from other node joins locally generated task and both are handled by equal priority.
- The task arrival process is assumed to be poisson and scheduling, communication and execution times are exponentially distributed.
- Local CPU serves on FCFS basis with no preemption.

Though the paper assumes homogeneity (each node has same processing capability), and symmetry (each node is linked to same number of nodes), we eliminate such assumptions. We will also attempt to simulate (for comparison purpose) semi-distributed control system (where only subset of nodes make scheduling decision), and the algorithms – bidding and nearest neighbor balancing on which this scheduling is based.

References

[1] Ishfaq Ahmad, Arif Ghafoor and Kishan Mehrotra ,"A Decentralized Task scheduling Algorithm & its Performance Modeling for Computer Networks", in *Proc. of the Third IEEE Symposium on Parallel and Distributed Processing,91*.