

COP5615 – Project2(README FILE)

Group Members : Prerna Prerna

Jyotsana Walia

Introduction

Over the last decade there has been a paradigm shift from centralized computation to highly distributed systems. The reasons for this change are mainly three. First, the users are very heterogeneous in nature and their participation may be inconsistent in the P2P network because of their own wish or a failure, which makes fault tolerance to be imperative in the system. Second, the approach of central coordination cannot does not work in P2P network because of the enormous number of users. Third, in large systems like sensor networks, the data is collectively is more accurate than data at single node. At the same time the size of the messages exchanged could be small to ensure efficient utilization of the scarce bandwidth available.

Description Of the Project

This project has implemented two gossip protocols – Gossip and Pushsum on topologies – Line, Full, 3D and imperfect 3D.

Protocols -

1. Gossip Protocol – In this the message propagates from a node to another neighbor node of that particular node. Each node receives a count of how many times it has received a rumor. When the final count reaches 20 the system shuts down.
2. Pushsum Protocol – This protocol can be described by the following algorithm.
 1. Each node starts with a sum $s(i)$ and a weight $w(i)$
 2. Initially $s(i)$ is value at i th node and $w(i) = 1$
 3. At each round, select a random node and send $[s(i)/2, w(i)/2]$ to it and $[s(i)/2, w(i)/2]$ to itself.
 4. At each round node i sends half of its values to someone and gets values and weights from someone else.
 5. Hope: It should have a mass conservation leading to sum of all $s(i)$ = total sum and sum of all $w(i)$ = n , Therefore average = $s(i)/w(i)$.The above algorithm can also be used to compute sum when the query node has $w=1$ and everyone else has $w=0$

Topologies-

1. Line – Every node has two neighbor nodes except for the first and the last node which have only one neighbor each.
2. Full – Every node is connected to every other node in the topology.
3. 3D – It has a 3D cube like structure in which according to the position of the node, it has different neighbors. The maximum neighbor that a node can have are 6.

4. Imperfect 3D – It is same as 3D with an addition of an extra neighbor node which is chosen randomly from all nodes except that particular node's neighbor nodes.

Implementation Details:

1. Project2.scala file – Contains the main class, the master class (networkMaster) and the worker class(WorkerNode).
2. Input : The input to the file needs to be given as the command line arguments in the following sequence –

Algorithm TotalNodes Toplogy

For example ---> Gossip 20 Line

Algorithm argument can have the following values-

- gossip
- pushsum

Topology argument can have the following values –

- line
- full
- 3d
- Imp3d

3. Expected Output: The output should show the convergence time achieved by a particular algorithm implementing a particular topology.

Maximum Network Topology for Gossip

- Line – 3375
- Full – 3375
- 3D – 3375
- Imperfect 3D - 729

Maximum Network Topology for Pushsum

- Line – 3375
- Full – 3375
- 3D – 3375
- Imperfect 3D – 1331

References

<http://www.phy.ornl.gov/csep/ca/node22.html> - which describes all the topology and neighbors

<http://www.cs.cornell.edu/johannes/papers/2003/focs2003-gossip.pdf> - paper by prof dobra

<http://stackoverflow.com/>

Extra technical learnings : Collections in scala and its operations, Maths package, Implicit Parameters