Report for Reading assignment on T-Man

# Motivation

*Why is the problem addressed in the paper interesting and important for the larger community to be solved?*

In a distributed environment, the communication topology plays a crucial important in performing functions such as searching, routing, information dissemination, also it contributes a lot to scalability and robustness to failures. While constructing and maintaining such a topology in dynamic, large-scale and fully distributed systems is extraordinary challenging. The communication topology must sustain the implemented application, it should reflect physical proximity and be robust, meanwhile, it shall be acceptable to construct and maintain such topology in a low cost.

# Contribution

*What are the main contributions of the paper?*

The paper defines topology management as an abstract service or a general purpose function desired in large distributed systems, which is independent from application. And the paper proposes a gossip-based probabilistic scheme called T-Man for constructing various different topologies, and such topology is defined by a ranking function that given a fixed node other nodes are sorted based on some appropriate property of components, such as geographical location proximity.

# Solution

*How did the authors solve the problem at hand?*

Each node in a routed network has an address to be able to send message, and it maintains addresses of others by a partial view, named a set of c node descriptors. Apart from the address, a node contains a profile, and is used by a ranking function to order a list of nodes in ascending distance from a given node.

The author exams three topologies: line and ring has a real number as a profile; mesh and torus has two-dimensional real vectors and profile; binary tree has binary string of length m as profile, and the ranking function is generated by d(x, y) which gives the distance between nodes x and y. Then each node runs an active thread and a passive thread, in active thread, a node select a peer, which is a random sample from the first half of the view in ranking order, and sends its view and descriptor as a buffer to that node, when it receives a view from other, it merge its current view with received view, and select first c elements in ranking order as its new view. Whilst the passive thread does the same thing as sending buffer and update view. Therefore nodes optimize their own views using views of close neighbors, which results in bring neighbors closer and closer as well as better utilization of views.

# Evaluation

*How good is the solution?*

*How did the authors evaluate their solution?*

*How good was the evaluation of their work?*

# Disadvantages of the Solution

*What are the disadvantages and shortcomings of the solution given by the authors?*

There are two error regarding to the deviations from rapid convergence model, one is called unbalanced contact distribution, a node initiates just one communication during T time units but it can be contacted several times which is actually a random number, for example, a node communicates too many times, it converges faster therefore provide less useful nodes to others, the same holds if a node communicates too few time. Secondly when two nodes exchange views after cycle i, the sample sets in two views are not from exactly the same distribution, because the closest c nodes that model refers to are nearest to the node which holds the view, even though we assume at all nodes views follow the model in a cycle i.

# Disadvantages of the Evaluation

*During the evaluation of their solution, did the authors overlook something?*

# Further improvements

*Are there any further improvements that can be made to the solution?*

In order to balance the number of contacts, nodes can count the total number of contacts they had, and only accept connection if this number is smaller than the current cycle number, as well nodes can search for neighbors from the view until a connection is accepted. To optimized endgame, SELECTPEER can assign to neighbors exponentially decreasing sampling probabilities according to increasing rank. Moreover, direction dependant ranking can be used for clustering and sorting, and synchronization and dynamism can be handled by periodic automatic restarting or organizing the runs into epochs.

*Are there any future directions you can think of?*

The solution assumes nodes have access to local clocks, or in other words, there exists a single synchronization point, so the model can be tailored so that it won’t require the protocol to be started at all nodes at the same time. Furthermore, it’s better to have mechanisms to ensure communication channels and the nodes are reliable.

# Answers of Questions

1. How would you construct the Chord topology using T-Man? Write your solution clearly and mention all the assumptions, advantages and disadvantages of your solution.

aaa