

CS541 - Operating Systems Assignment # 2

Introduction

For this assignment, you are going to implement *vector logical clocks*. Each node will send messages to the other nodes; the messages will include the node's view of "time" as it knows it. The receiving node will then update its view of time at the other nodes from the information received in the message. We noted that one node's notion of time at the other nodes improves when more messages are sent. This assignment will explore that notion.

For this assignment, each node will send messages to other random nodes (i.e., the destination node will be selected randomly). The message will consist of the sending node's vector clock values. The receiving node will update its vector with the information that has been received. In order to see the effect of "busy" nodes (those sending a lot of messages) vs "non-busy" nodes, the times at which nodes send messages should be random, but based on a distribution. One way to do this is to vary the sending times of each node, determined by its rank. On average, the root node should send out one message per unit time, node with rank one should send out $\frac{1}{2}$ per unit time, node with rank two should send out $\frac{1}{4}$ per unit time, etc. That is, the number of messages a node sends should be related to its rank n as $\frac{1}{2^n}$.

Run the experiment long enough so that several hundred messages are sent, then compare the vector clocks at each node. Comment on the trend that you see in the accuracy of the clocks at each node.

You should use MPI to implement this program.