Verification of the Statistical Model for Multi-Stage Message Distribution

Jedi High Council

July 11, 2013

1 Monte Carlo Simulation

To verify our model we implemented a monte carlo simulation that simulates the behavior of the network given parameters n, m, and k. In particular, the simulation initializes the network of n nodes in an unconnected state (i.e. all nodes are disconnected from each other and there exists a single node, the key manager, who possesses the key). Then, in discrete time steps, the simulation attempts to establish new connections between a pair of nodes u and v with probability p_1. In addition, nodes already in the process of establishing a connection proceed forward through the m stages of message distribution by a single step with probability p_2. At the end of one time step, the overall discrete time is incremented by one, and the process repeats until all nodes have been connected and received the key. This procedure is formalized in Algorithm 1. The source code, written in Matlab, has been made available at ¡LINK¿ for interested readers to experiment with. It is a self contained program that contains all of the required documentation within.

We repeated this general procedure for $T = 10000$ iterations each for different values of n, k, and m, collecting the average time over all program runs, standard deviation of each run, and the estimated error in the simulation. Our estimated error was quite small and well within the expected bounds of precision. We then computed the difference Δ between the output from our model and the average time reported by this monte carlo simulation to verify the correctness. We found that for all configurations considered the delta was always less than 0.2. We attribute this difference to the error

introduced in monte carlo simulation.

Data: T, k, m, n, p_1, and p_2
Result: Expected time
$A_c \leftarrow zeros[1 \dots n][1 \dots n]$;
$A_m \leftarrow zeros[1 \dots k][1 \dots n][1 \dots n]$;
$n_c \leftarrow 0$;
$C_l \leftarrow zeros[1 \dots n]$;
$total \leftarrow 0$;
for $T_i = 0 \rightarrow T$ **do**
  $t \leftarrow 0$;
  while $n_c < n - 1$ **do**
    Build a list of candidate child nodes ready to receive a new message (i.e. those unconnected and not receiving a message already). Filter the list by randomly discarding each candidate node with probability $1 - p_1$ **for** $m_i = 0 \rightarrow m$ **do**
      With probability p_2, advance each child node in stage S_{m_i} to $S_{m_{i+1}}$. If a node advances to stage S_m, set them as connected in C_l, update their connection with the parent in A_c, and discard their message trace in A_m.;
    end
    Randomly assign each child node in the message ready list to an available parent. If the number of available parents is less than the number of ready children, then a subset of those in the ready list begin a communication trace. Else, every children begins a communication trace.;
    $t \leftarrow t + 1$;
  end
  $total \leftarrow total + t$;
end
output $total/T$

Algorithm 1: Monte carlo simulation to verify the statistical model