**TCP/IP Mini-project**

1. **Estimate the distance between RFID tag installations when trains approach from opposite directions**
   * **Compute the distance for 100kmph each**

The trains are usually allowed a maximum deceleration of 5-15% of g (acceleration due to gravity). Hence, considering average deceleration to be 0.1g,

* the braking distance of train = u^2/2a = 0.385 km.
* the braking time = u/a = 28 seconds.

Assume train to server and server to train data transfer time = 0.5 s each.

Thus, Round trip time (RTT) of data packets = 1s.

Total time for sending and receiving information for train = RTT + T\_s (server processing time).

Assume T\_s = 1s.

Distance travelled by each train in time RTT + T\_s = 56 m.

Minimum distance between trains for a safe stop with brakes = 56 + 385 + 385 + 56 = 882 m.

So if the tags are placed 1 km apart and the train decide to stop when they are at 3km apart, server can send a signal at 3km distance apart.

Optimal distance between tags = 1km. Additional tags serve as failsafe options. Also additionally train length is taken to be 1 km.

1. **Estimate the latency from sensing to dashboard**
   * **Train 2’s position is known on Train 1s dashboard**

Use ping command to get the latency.

1. **Estimate the reliability of the system under packet errors**
   * **How much of bit error can be tolerated?**

No bit error can be tolerated as no error correction algorithms are use.

1. **Compare latencies for a few connection scenarios**
   * **one-to-one connection between trains, train and server, train-server to server-train forwarding**

Use ping again.

1. **Ensure only data from the specific trains are accepted to the server network**

Use SSL.

1. **Ensure the system is fail-safe**

Multiple tags and justification.

Server handling not getting info from trains. Even in case train stops, server will take it to be stopped at that location. Handles reverse of train too. And collision avoidance from back. Stop signal from server at 3km even if braking distance is 500 m each at most.

TCP protocol of retransmission in case of packet loss.