**Executive Summary**

This report presents simulations focused on packet transmission through single-link and compound networks. The goal was to estimate the average number of transmissions required for an application message consisting of different numbers of packets, under various packet failure probabilities. The key output includes graphs that show how the number of transmissions increases as the probability of packet failure increases, offering insight into the network’s reliability.

For Task 4, a compound network with uniform failure probabilities across all links was simulated. The results reveal that as the probability of failure increases, so does the number of transmissions required, with interesting non-linear growth patterns for larger message sizes. For Task 5, where each link was given different probabilities of failure, the results highlighted how imbalances in link reliability influence overall network performance. These simulations provide valuable insights into network behavior under varying conditions and suggest that ensuring consistent link performance across the network can significantly reduce transmission overhead.

**Code Repository**

The simulation code is stored in a repository that includes functions to simulate single-link, parallel, and compound networks. To access and run the programs:

* Clone the repository: git clone <repository-link>
* Navigate to the directory: cd network-simulations
* Run the simulations via MATLAB: Use runCustomCompoundNetworkSim() for the compound network task, with the required K values and probabilities.

**Task 1 – Introduction to Single-Link Networks**

**Objective**: Simulate the average number of transmissions required to transmit K packets through a single link, where each transmission has a probability ppp of failing.

**Explanation**: For each packet, the transmission is attempted repeatedly until it succeeds. The number of attempts required follows a geometric distribution, as the success of each packet transmission is independent and occurs with probability 1−p1 - p1−p. The code simulates this process for 1000 iterations per value of ppp, and averages the results.

**Key Code Snippet**:

matlab

for iter = 1:numIterations

success = false;

while ~success

if rand() >= p

success = true;

end

transmissions = transmissions + 1;

end

end

**Task 4 – Simulating a Compound Network**

**Objective**: Simulate packet transmission through a compound network consisting of two links in parallel. Both links have the same failure probability ppp, and a packet is successfully transmitted if either link succeeds.

**Explanation**: The compound network operates in parallel, meaning each packet can pass through either link. The transmission is only considered failed if both links fail simultaneously. The simulation checks for a success if either of the two links transmits successfully, with failure happening only if both fail.

**Interesting Characteristics of Output**: As ppp increases, the average number of transmissions increases significantly. However, the network's parallel structure makes it more resilient to failure compared to the single-link network. The redundancy of having two paths allows for better reliability, particularly when ppp is low to moderate. For higher values of ppp, the growth in transmission attempts is more pronounced, showing a non-linear relationship as the failure rate approaches 1.

**Key Code Snippet**:

if rand() >= p || rand() >= p

success = true;

end

**Graph Commentary**:

* For small K values, the growth in the number of transmissions is more moderate.
* For larger K values (e.g., K = 100), the transmission attempts increase rapidly as ppp approaches 1. This reflects how larger messages are more susceptible to compound failures, making the network less efficient at higher failure probabilities.

**Task 5 – Custom Compound Network with Different Probabilities**

**Objective**: Extend the compound network simulation to allow different probabilities for each link. In this task, the network consists of three links in series, with different failure probabilities assigned to each.

**Explanation**: The custom compound network introduces diversity in link reliability, where each link (link 1, link 2, and link 3) has a unique failure probability. A packet is considered successfully transmitted if it passes through the network without failing at any of the three links. This simulation calculates the number of transmissions needed, depending on the balance between the most and least reliable links.

**Key Code Snippet**:

if rand() < p1 || (rand() >= p2 && rand() >= p3)

success = true;

end

**Interesting Characteristics of Output**: The variability in link probabilities leads to fascinating behavior in the number of transmissions. When one of the links has a very low failure probability, the overall performance of the network significantly improves, regardless of the failure rates of the other two links. On the other hand, if the most unreliable link has a high probability of failure, the overall network performance declines rapidly.

For example, when p1=0.10p\_1 = 0.10p1​=0.10, p2=0.60p\_2 = 0.60p2​=0.60, and p3p\_3p3​ varies, the network performs well at low p3p\_3p3​, but as p3p\_3p3​ increases, the number of transmissions spikes considerably. This highlights the criticality of ensuring that all links in the network are reasonably reliable for the system to function efficiently.

**Graph Commentary**:

* Figures show a consistent trend where imbalanced link probabilities make the network performance highly sensitive to the worst-performing link.
* When p1p\_1p1​ and p2p\_2p2​ are kept low, the overall performance is stable even with large K values, but high probabilities on any one link degrade performance significantly.

**Conclusion**

These simulations reveal key insights about network reliability:

* **Task 4**: Compound networks with uniform probabilities demonstrate improved reliability over single-link networks, thanks to redundancy. However, performance worsens rapidly when the failure probability is high.
* **Task 5**: Imbalances in link probabilities disproportionately affect the overall network performance. Ensuring that no single link dominates the failure rate is crucial for maintaining an efficient network.

These results can inform the design of more reliable communication networks, where redundancy and consistent reliability across all links can help minimize transmission attempts.