CS-521 Term Project: Packet vs. Circuit Switching

Lucas Ferreira, Vishal Rao, Justin Ho

29 April 2021

We pledge our honor that we have abided by the Stevens Honor System.
Lucas Ferreira, Vishal Rao, Justin Ho

1 Introduction

Packet-switched networks break streams of data into small blocks known as packets. Each of these packets are then sent independently over a shared network, based on the destination address in each packet. After receiving, packets are reassembled in the proper sequence to make up the message. Data is processed at all intermediate node including source system. The delay between data units is not uniform. It is suitable for handling bilateral traffic. And, multiple users can use the same channel while transferring their packets. If one route is failed the data can be routed from other paths too.

Circuit-switched networks require dedicated point-to-point connections. The resources needed are reserved for the duration of the communication between the end systems. It was designed specifically for voice communication and packet switched networks handled data. Data is processed only at source system. And, the delay between data units is uniform. Circuit switching is not convenient for handling bilateral traffic. It is preferred when the communication is long and continuous.

To compare the performance of packet-switched and circuit-switched networks, we created an event-based simulation model for either network. A series of experiments were designed to test how either network reacts to packets being sent across them, and whether one has an overall better performance than the other.

2 Analysis

2.1 Methodology

To compare and contrast the performance of packet switching and circuit switching, two event-based simulation models written in Python were created to simulate a packet switching network and a circuit switching network. The intent of creating these models was to provide numerical insight into how the time data packets spend traveling through a network is affected by the effects of packet switching and circuit switching. By generating statistics such as average data packet queue delay and average data packet total delay for a network simulation, a strong basis for comparing the two types of networks was formulated.

For both simulations, a simple bi-directional network graph consisting of six nodes was used, as shown in Figure 2.1, where fixed-size data packets are randomly generated at node 0, and sent through the network where their destination is always node 5. The models created allow for the mean arrival rate of each new data packet, the total simulation duration, and the transmission time between nodes to be manually set. For example, a circuit switching simulation could be run with a new packet mean arrival time of 500 milliseconds, a total simulation duration of two hours, and a transmission time between nodes of 80 milliseconds. Note that the transmission delay at the network nodes is disregarded. At the end of each simulation, the results, which include average queue delay and total delay statistics, are printed to the console.

The circuit switching model simulates the way circuit switching networks are implemented by dedicating an entire channel through the network from source to destination when a data packet is ready to be transmitted. Whenever new data packets are generated and cannot be transmitted because the there are no available channels, they enter a queue where they remain until new channels become available. This process

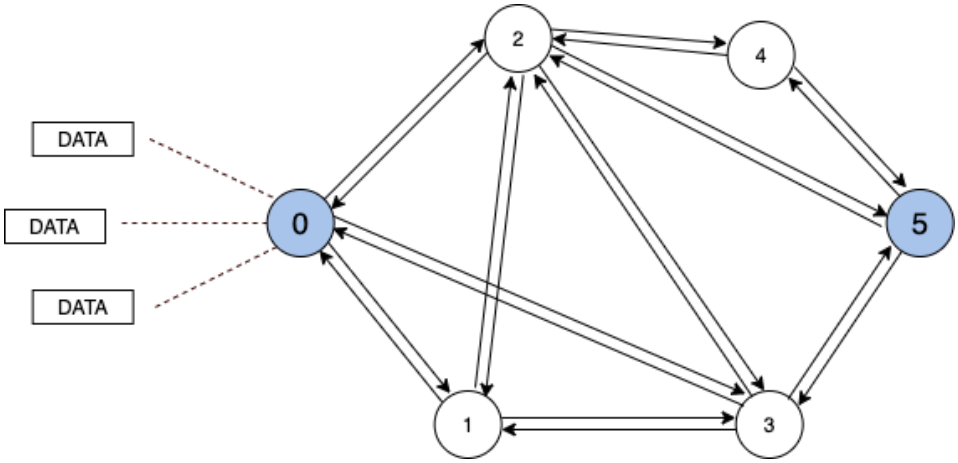


Figure 2.1: Network graph used in simulation model

is illustrated in Figure 2.2. The packet switching model has a different mechanism for handling channels when data packets are being transmitted through the network. Rather than securing the entire channel from a data packet's destination node to its source node, the packet switching model only secures individual channels between two different nodes as a data packet moves throughout its assigned path. For example, if a data packet being transmitted from node 0 to node 5, which has been assigned the path 0-1-2-5, is traveling through the 1-2 channel, only the 1-2 channel is secured, and thus blocked. Whenever a data packet arrives at a new node, it checks if the next channel it needs to use is available. If it isn't, it searches for a new channel which can lead the data packet to its destination. If no other channels are available, the data packet is inserted into a queue belonging to the last node it visited. As new channels become available, the data packet is popped from the queue and resumes its trip to its destination. The packet switching process is illustrated in Figure 2.3.

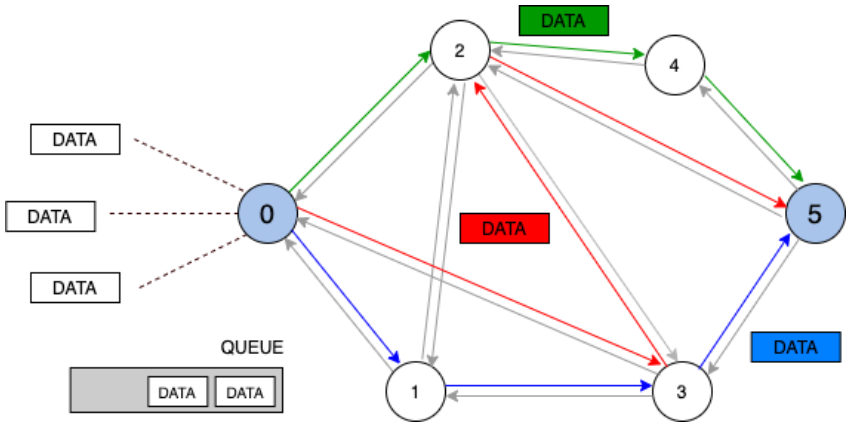


Figure 2.2: Circuit switching model

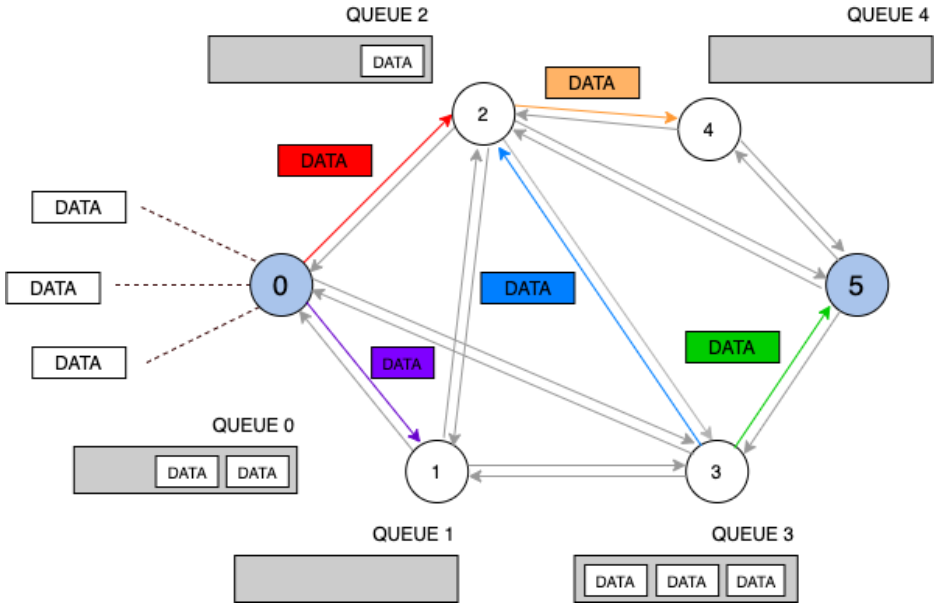


Figure 2.3: Packet switching model

2.2 Graphs and Tables

All time measurements are in milliseconds.

Table 1: Circuit Switching Simulation Results

Transmit Time	Avg. Arrival Time	Packets Arrived	Packets Completed	Avg. Total Delay	Avg. Queue Delay
20	42	85,825	85,821	43.36	0.3
	54	66,553	66,551	42.17	0.12
	66	54,428	54,426	41.48	0.06
	78	46,126	46,125	41.09	0.03
	90	39,815	39,814	40.8	0.02
30	42	85,668	85,666	70.03	2.14
	54	67,008	67,006	66.56	0.77
	66	54,354	54,352	64.55	0.33
	78	46,304	46,303	63.51	0.2
	90	40,209	40,207	62.68	0.11
40	42	85,680	85,678	105.92	12.0
	54	66,557	66,553	94.26	3.31
	66	54,436	54,435	89.94	1.51
	78	46,193	46,191	87.49	0.71
	90	40,027	40,026	85.85	0.42
50	42	85,511	85,509	232.07	112.28
	54	66,922	66,919	130.07	13.05
	66	54,719	54,715	118.76	4.78
	78	46,151	46,149	113.65	2.21
	90	39,984	39,983	110.78	1.27

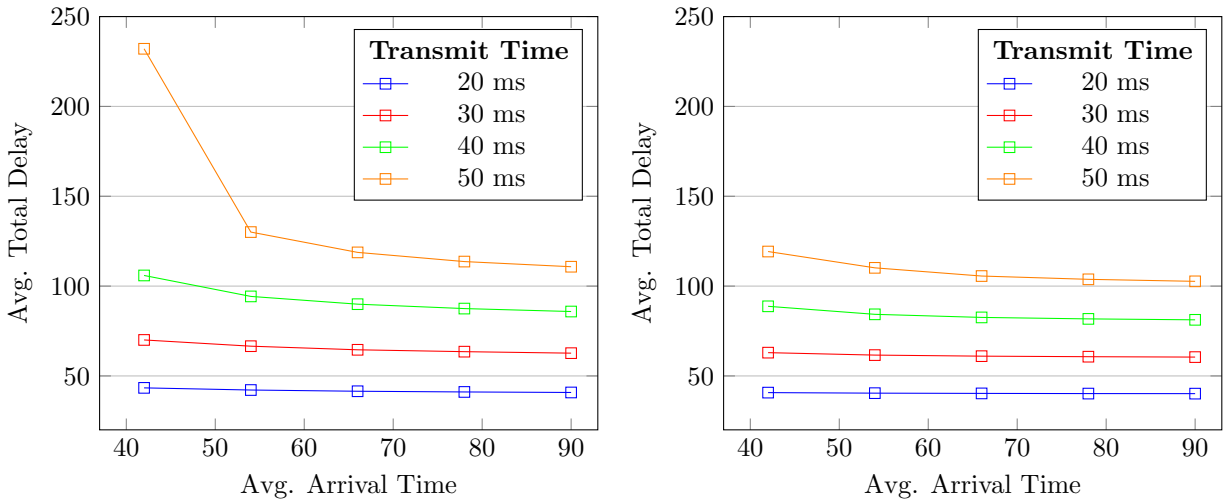
3 Code and Output

Source code and code output have been attached to this report: see `out.txt` for output generated for Tables 1 and 2 and Figures 2.4 and 2.5; see `out-40ms-MAT.txt` for output generated with a 40 ms mean arrival time; see the Python files for source code for the simulations.

Table 2: Packet Switching Simulation Results

Transmit Time	Avg. Arrival Time	Packets Arrived	Packets Completed	Avg. Total Delay	Avg. Queue Delay
20	42	85,627	85,625	40.72	0.03
	54	66,748	66,746	40.42	0.01
	66	54,707	54,705	40.28	0.01
	78	46,332	46,331	40.18	0.01
	90	39,849	39,848	40.15	0.01
30	42	85,741	85,739	62.96	0.21
	54	66,621	66,619	61.6	0.09
	66	54,514	54,513	61.01	0.04
	78	46,247	46,245	60.71	0.02
	90	39,977	39,975	60.49	0.01
40	42	85,516	85,512	88.75	0.82
	54	66,675	66,673	84.29	0.33
	66	54,294	54,291	82.57	0.14
	78	46,128	46,126	81.75	0.08
	90	39,969	39,966	81.23	0.04
50	42	85,858	85,855	119.24	2.05
	54	66,559	66,555	110.16	0.95
	66	54,308	54,306	105.6	0.42
	78	46,172	46,170	103.77	0.23
	90	40,030	40,029	102.65	0.12

Figure 2.4: Circuit Switching vs. Packet Switching – Avg. Arrival Time



(a) Circuit Switching

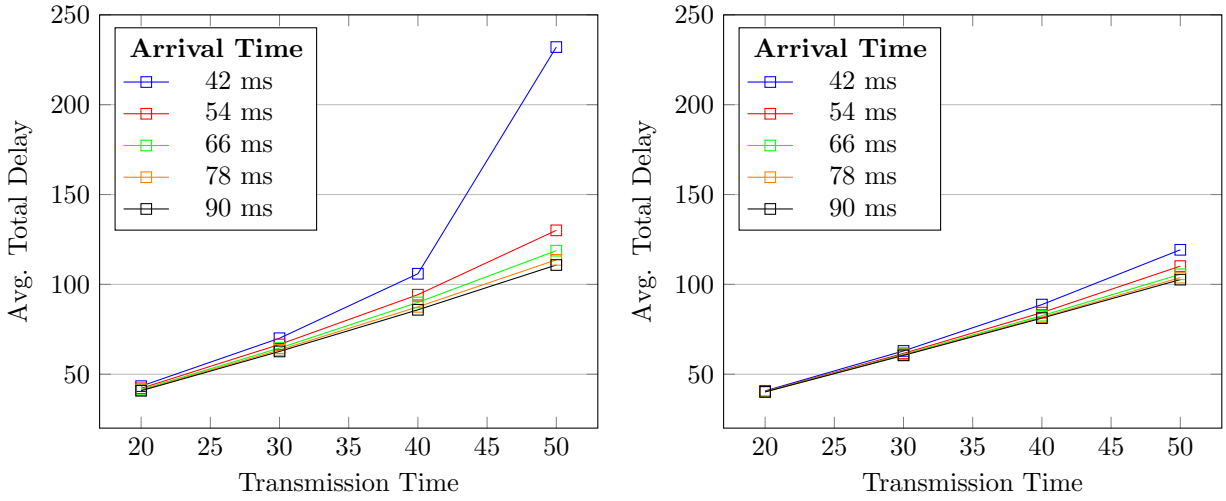
(b) Packet Switching

4 Observations and Recommendations

We contrast the efficacies of both switching methods in Figures 2.4 and 2.5. In the figures, we observe how each method scales in different configurations of transmission time and average arrival time. In Figure 2.4, we set the average total delay as a function of the average packet arrival time, with each line representing a fixed transmission time, from 20 ms to 50 ms with 10 ms jumps. Comparing the performance of circuit and packet switching, there is a clear efficiency disparity; while packet switching stays relatively constant with the arrival-transmission time configurations, circuit switching shows extreme service degradation as transmission time grows larger than the average arrival time, with the 42 ms-50 ms arrival-transmission configuration having more than two times the average total delay than the 42-40 configuration and almost two times the total delay than the 54-50 configuration. This observation is further reinforced by the results from `out-40ms-MAT.txt`: the 40-50 configuration has an average total delay of over 4,000 ms, greater than the 42-50 configuration by a factor of nearly 20.

In Figure 2.5, we instead set the average total delay as a function of transmission time. Similarly to

Figure 2.5: Circuit Switching vs. Packet Switching – Transmission Time



(a) Circuit Switching

(b) Packet Switching

Figure 2.4, we see that circuit switching shows great service degradation; the 42-50 configuration follows an interpolated exponential curve, while every other configuration follow interpolated linear curves.

From the observed results, it is clear that as the arrival-transmission ratio tends to 0—in other words, as the transmission time tends to a value greater than the average arrival time —performance tends to degrade using both methods. However, circuit switching demonstrates much worse degradations, as having longer transmission times under the circuit switching model specifically implies that the transmission channels the packets travel through must stay dedicated to transmitting their respective packets for longer, resulting in exorbitantly large packet queues, filled with packets actively waiting for when a route to their destinations opens up. This problem is largely mitigated under the packet switching model as packets do not need an entire route being open to be transmitted; instead, packets only need an (acyclic) open router node towards their destinations.

5 Conclusion