Chi Kei Loi, ID:912371115

Tingting Zhu, ID:913356317

ECS 152A

Project2 part2

* 1. **A Sample Execution**

L: The number of packets in the queue.

N: The number of times the packet at the head of the queue has been retransmitted. When a new packet comes to the head of the queue n is reset to 0.

S: The slot number when the next transmission attempt will be made for the packet at the head of the queue.

|  |  |  |
| --- | --- | --- |
| Host1 | Host2 | Host3 |
| L =2 | L =3 | L =2 |
| S =100 | S =100 | S =100 |
| N =0 | N= 0 | N=0 |

There is a collision between Host 1, 2 and 3 in slot 100.

|  |  |  |
| --- | --- | --- |
| R1 = 1\*0.6621015449=1 | R2=1\*0.0598288625=0 | R2=1\*0.5979568986=1 |

Retransmissions schedule.

|  |  |  |
| --- | --- | --- |
| L= 2 | L =3 | L =2 |
| S = 102 | S = 101 | S =102 |
| N = 1 | N =1 | N = 1 |

In slot 101, host2 transmits packet.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 2 | L = 2 |
| S = 102 | S = 102 | S = 102 |
| N = 1 | N = 0 | N = 1 |

In slot 102, collision between host 1, 2 and 3.

|  |  |  |
| --- | --- | --- |
| R1 = 2\*0.0940302045=0 | R2=1\*0.9731501949=1 | R2=2\*0.0662842158=0 |

Retransmissions schedule.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 2 | L = 2 |
| S = 103 | S = 104 | S = 103 |
| N = 2 | N = 1 | N = 2 |

Collision between host 1 and 2 in slot 103.

|  |  |  |
| --- | --- | --- |
| R1 = 4\*0.5671123588=2 |  | R2=4\*0.2960073405=1 |

Retransmissions schedule.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 2 | L = 2 |
| S = 106 | S = 104 | S = 105 |
| N = 3 | N = 1 | N = 3 |

Now host2 transmit packet in slot 104.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 1 | L = 2 |
| S = 106 | S = 105 | S = 105 |
| N = 3 | N = 0 | N = 3 |

Collision between host 2 and 3 in slot 105.

|  |  |  |
| --- | --- | --- |
|  | R2 =0.0836140907 = 0 | R3 = 8\*0.1982544372 = 2 |

Retransmissions schedule.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 1 | L = 2 |
| S = 106 | S = 106 | S = 108 |
| N = 3 | N = 1 | N = 4 |

Collision between host 1 and 2.

|  |  |  |
| --- | --- | --- |
| R1 =8 \* 0.3973937067 =3 | R2 =2 \*0.1825762354= 0 |  |

Retransition schedule.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 1 | L = 2 |
| S = 110 | S = 107 | S = 108 |
| N = 4 | N = 2 | N = 4 |

Host 2 transmit packet to slot 107.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 0 | L = 2 |
| S = 110 | S | S = 108 |
| N = 4 | N = 0 | N = 4 |

Host 2 has no packet to transmit, host 3 transmits packet to slot 108.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 0 | L = 1 |
| S = 110 | S | S = 109 |
| N = 4 | N = 0 | N = 0 |

Host 3 transmits packet again.

|  |  |  |
| --- | --- | --- |
| L = 2 | L = 0 | L = 0 |
| S = 110 | S | S |
| N = 4 | N = 0 | N = 0 |

Host 3 finishes transmitting packets, host 1 is the only active host now. Host 1 transmit packet in slot 110.

|  |  |  |
| --- | --- | --- |
| L = 1 | L = 0 | L = 0 |
| S = 111 | S | S |
| N = 0 | N = 0 | N = 0 |

Host 1 still has one more packet to transmit.

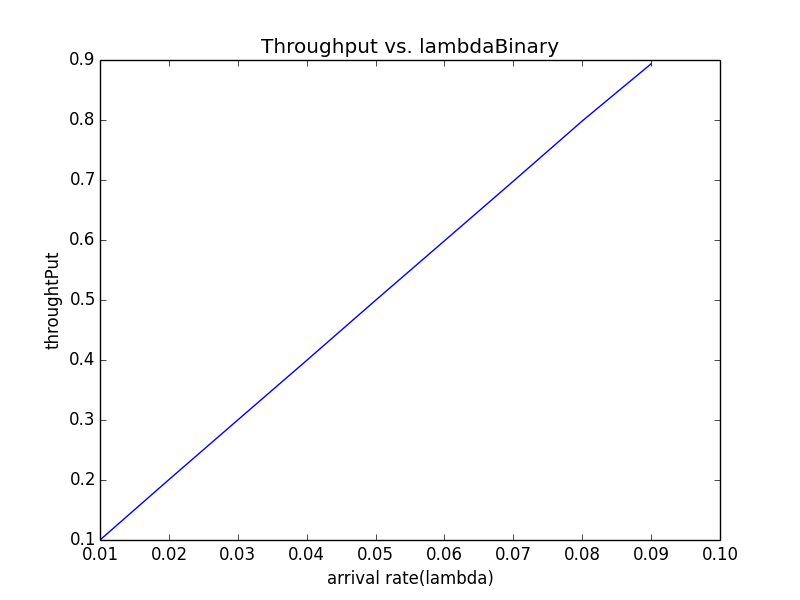
|  |  |  |
| --- | --- | --- |
| L = 0 | L = 0 | L = 0 |
| S | S | S |
| N = 0 | N = 0 | N = 0 |

All packets finish transmitting in time slot 111.

* 1. **Simulation analysis**

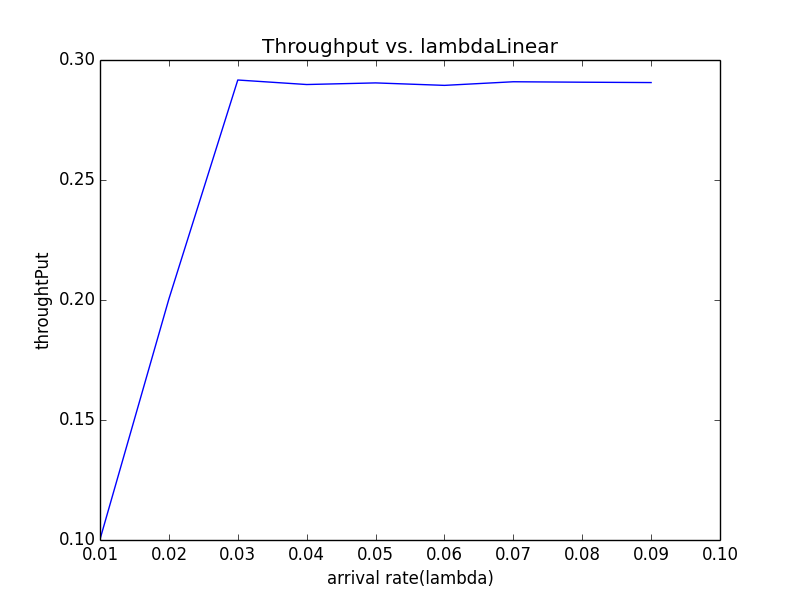
(1) Binary exponential backoff algorithm:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lambda | Throughput | Successes | Collisions | Idle |
| 0.01 | 0.10005 | 100049 | 9035 | 890915 |
| 0.02 | 0.20061 | 200608 | 47280 | 752111 |
| 0.03 | 0.30005 | 300050 | 136274 | 563675 |
| 0.04 | 0.39931 | 399305 | 194225 | 406469 |
| 0.05 | 0.49950 | 499502 | 173033 | 327464 |
| 0.06 | 0.59802 | 598641 | 132448 | 268910 |
| 0.07 | 0.69802 | 698019 | 94552 | 207428 |
| 0.08 | 0.79834 | 798338 | 62899 | 138762 |
| 0.09 | 0.89362 | 893622 | 35487 | 70890 |



(2) Linear backoff algorithm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lambda | Throughput | Successes | Collisions | Idle |
| 0.01 | 0.09991 | 99907 | 14885 | 885207 |
| 0.02 | 0.20015 | 200149 | 85377 | 714473 |
| 0.03 | 0.29052 | 2990518 | 617762 | 91719 |
| 0.04 | 0.29025 | 290251 | 632891 | 76857 |
| 0.05 | 0.29056 | 290561 | 632151 | 77287 |
| 0.06 | 0.8923 | 289234 | 633573 | 77192 |
| 0.07 | 0.29027 | 290272 | 632732 | 76995 |
| 0.08 | 0.28995 | 289953 | 632886 | 77160 |
| 0.09 | 0.29048 | 290481 | 632032 | 77486 |



Analysis of the result:

From the graph based on the data gathered from the simulation, we can see that the throughput of the binary exponential algorithm linearly increase while the arrival rate increase. In other word, the binary exponential backoff algorithm is more efficient.

For linear backoff algorithm, the throughput linearly increase up to lambda is 0.3, then remain the same for the rest of the lambdas.

The interval of random numbers in binary exponential algorithm gets wider while number of collisions increase. Thus, when collision occurs, the chance that two or more hosts are assigned to the same slot is lower with different interval since the collision number is different.

Thus, the number of slots we choose for different host have a greater chance to be different, which lead to less collisions and better utilization.

However, the linear backoff algorithm, we choose the r in a fixed range, then when lambda increase—packets send in the same slots increase/collision increase, the slots the algorithm choose will have a larger chance to be the same. Thus, the throughput cannot be increased because of high collisions rate.