PROJECT 5 - SWITCH PERFORMANCE INCLUDING BUFFERING

- 8541324935 -

Tamoghna Chattopadhyay

1. INTRODUCTION

The project builds up on the previous project and includes monitoring the buffer and queues on the input side. The traffic pattern in terms of desired output ports remain the same as in the previous project, we will also simulate the arrival of packets to the input ports.

A new packet will arrive to the input port in a slot with probability Parrival. This packet arrival probability is same for each input port. Each input port has a HOL slot and an additional number of buffers to store arriving traffic. Buffers are not shared between ports. If a packet arrives to find all the buffers for that input queue occupied, the packet is dropped. The number of dropped packets are recorded. To avoid excessive buffer overflows, the rate of packet arrival shouldnot exceed the maximum throughput capability of the switch.

Our focus is primarily on an 8X8 switch, with two types of traffic:

(i) balanced traffic : $\alpha_j = \frac{1}{N} \quad \forall j$

(ii) hot spot traffic : $\alpha_1 = \frac{1}{k}$

$$\alpha_j = \left(\frac{1}{N-1}\right)\left(\frac{k-1}{k}\right) \text{ for } j \neq 1$$

We look at cases $k = 2, 3$ and 8.

2. THEORY

The buffer occupancy for each input on a slot by slot basis can determine the 'steady state' queue size distribution and the mean queue length.

Little's Result states $N = \lambda T$
where,

N = average number in a system
λ = arrival rate in packets per unit time
T = time an average packet spends in system.

We also have to moniter the packet drops and number of packet delays while in the HOL slot due to HOL output port blocking for each output port.

8. Little's Result is very useful because of its generality.

- Nothing is assumed about the system
  - any part of the system can be considered as a black box

- The arrival process can be anything
  - in particular, one need not assume it to be a Poisson process
  - The process, however, has to be stationary.

3. SIMULATION METHODOLOGY

For the simulation, we first generate a random number to check whether the packet has arrived at a queue or not.

Once we verify that a packet has indeed come in, we generate a packet in the Buffer's 1st position with the probabilities:

$$\alpha_1 = \frac{1}{k}$$

$$\alpha_j = \left(\frac{1}{N-1}\right)\left(\frac{k-1}{k}\right) \qquad \forall\, j \neq 1$$

This is done by simulating that if a random number is less than $\frac{1}{k}$, we generate packet destined for output port 1 whereas when it's greater than $\frac{1}{k}$, we can generate any other packet with an equal probability. So for that, we can use,

randi ([2, N], 1, 1)

In this same loop, we can check whether the Buffer is empty before filling it up. If it is not empty, the packet is dropped and the count of dropped packet is updated.

Then we create a Hash table which indicates how many of the output ports are unique. And then, using this Hash, we create another Hash of random number to decide how the packets leave from the buffer when they are to be sent to the same output port.

How this works is that every packet from the buffer will access its position in the hash. If the

value in the Hash is 0, then it will continue
If it sees that the value in the Hash is 1,
it will send out the packet. Here, we keep
count of the number of packets sent out.
In the meantime, when we send out a packet,
we move the packets in the Buffer, and add
a empty spot represented by 0 in the end
of the Buffer for a new packet to arrive.

In the case of any other value in the Hash, the
Value in the Hash is decreased by 1.

After the packets are moved, we see the
number of non zero terms in the Buffer
which represents the packets. This gives
us the size of the input buffer each
time.

4. RESULTS AND OBSERVATIONS

- For Balanced Traffic, p = probability of arrival of packet

When $p = 0.5$,

No of packets dropped = 4
No of packets passed = 4030
Buffer size = 10

When we increase the buffer size to 20 or 30,
the number of packets dropped decreases,
often coming to 0.

When we increase the value of the probability of arrival of packet, then, when,

$p = 0.6$,

No. of packets dropped = 65
No. of packets passed = 4756
Buffer size = 10

But when we increase both the p and the buffer size, the amount of packets dropped decreases.

- For Hot Spot Traffic,

when $k = 2$, $p = 0.5$, buffer size = 10

No. of packets dropped = 1907
No. of packets passed = 2001

When we increase value of p in this case,
$k = 2$, $p = 0.6$, buffer size = 10

No. of packets dropped = 2614
No. of packets passed = 2071

So, we see that the number of packets dropped increases.

When we increase buffer size,
$k = 2$, $p = 0.5$, buffer size = 20

No. of packets dropped = 1793
No. of packets passed = 2042
So, we see that the number of packets dropped decreases.

When we increase the value of k to 3 keeping $p = 0.5$ and buffer size $= 10$, we see,

No of packets dropped $= 914$
No of packets passed $= 2958$

So, the no. of packets dropped decreases.

But as we increase p, the no. of packets dropped increases.

Similarly, as we increase buffer size, the no. of packets dropped decreases.

So, we see the same characteristics as in the case $k = 2$, except no. of packets dropped is less in case of $k = 3$.

The value for $k = 8$ is same as a balanced traffic.

Using Little's Formula, calculating the shaded area, i.e., the area under the curve and then dividing it by the total time, where $t = 1000$,

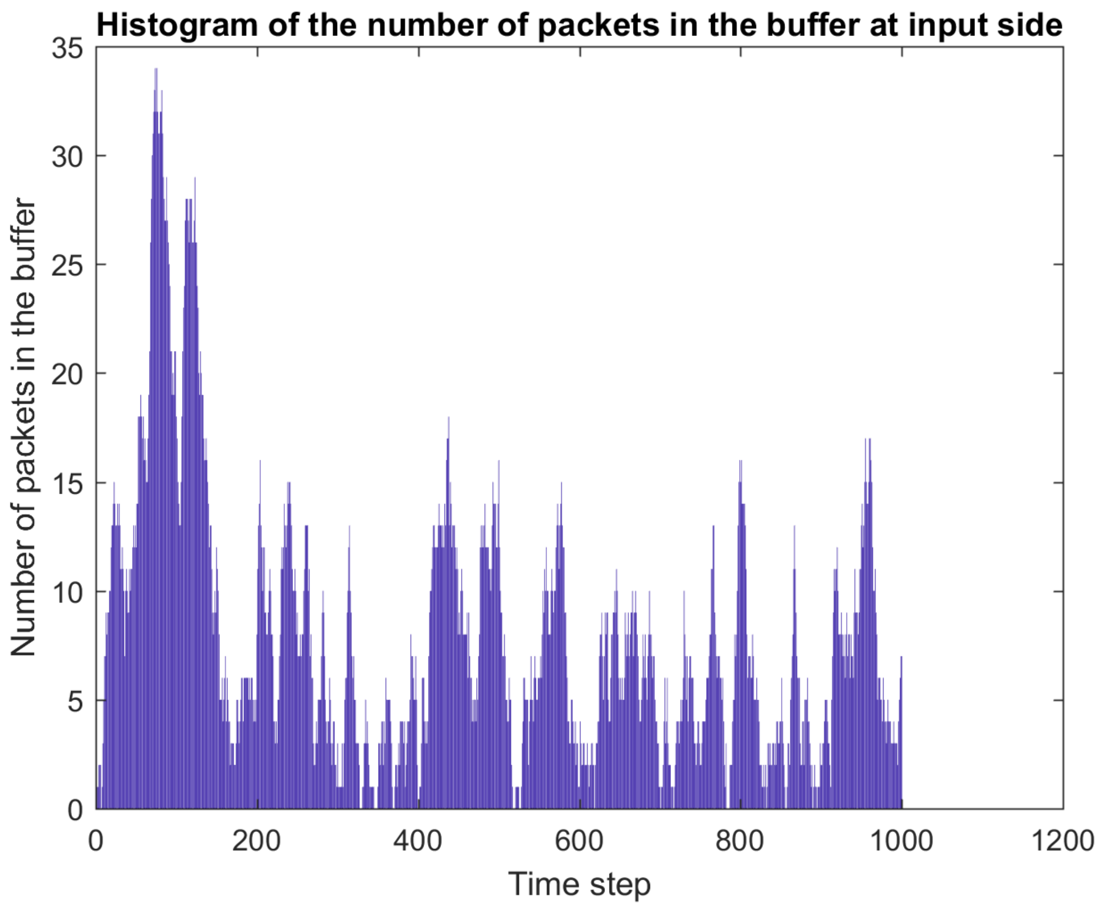
we can see, for eg.
$k = 3$, Buffer size $= 20$ $p = 0.5$
then,

$$\frac{\text{sum}}{t} = \frac{122294}{1000} = 122.294$$

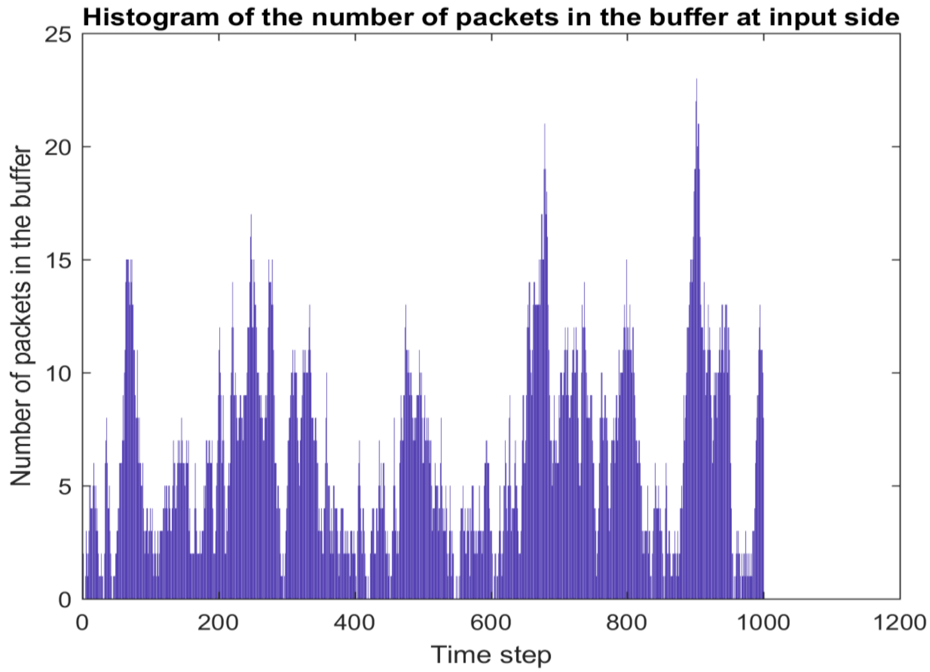
This is the mean queue length. If we know $\lambda = 10^6$ ms, then, $T = N/\lambda$ where $N = 122.294$.

This can be done for all the values of k, packet arrival probability and Buffer size.

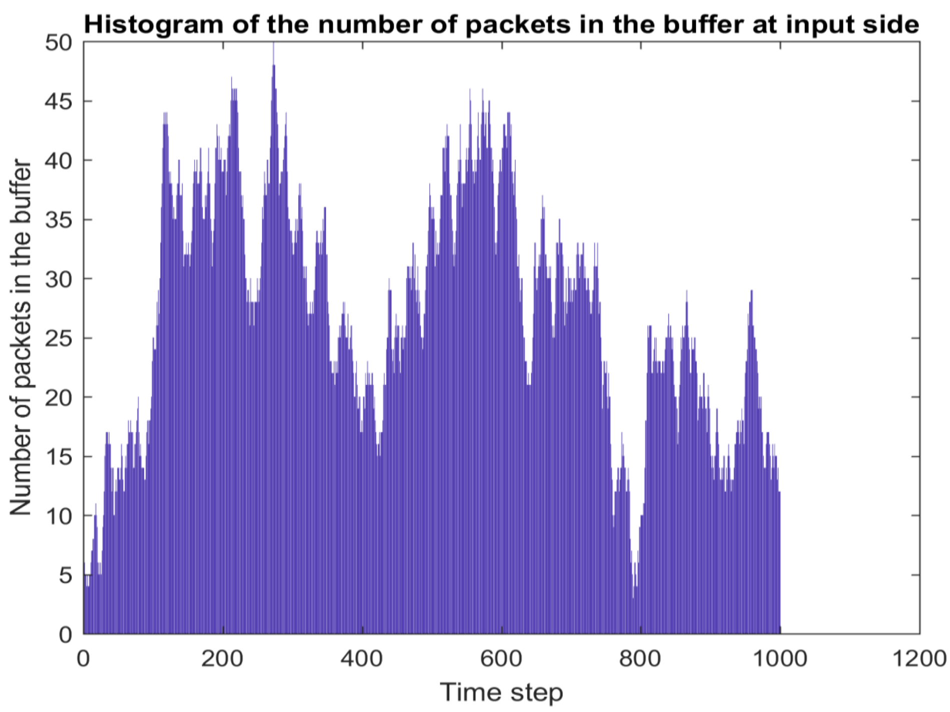
Following are some of the histograms got from the simulation:



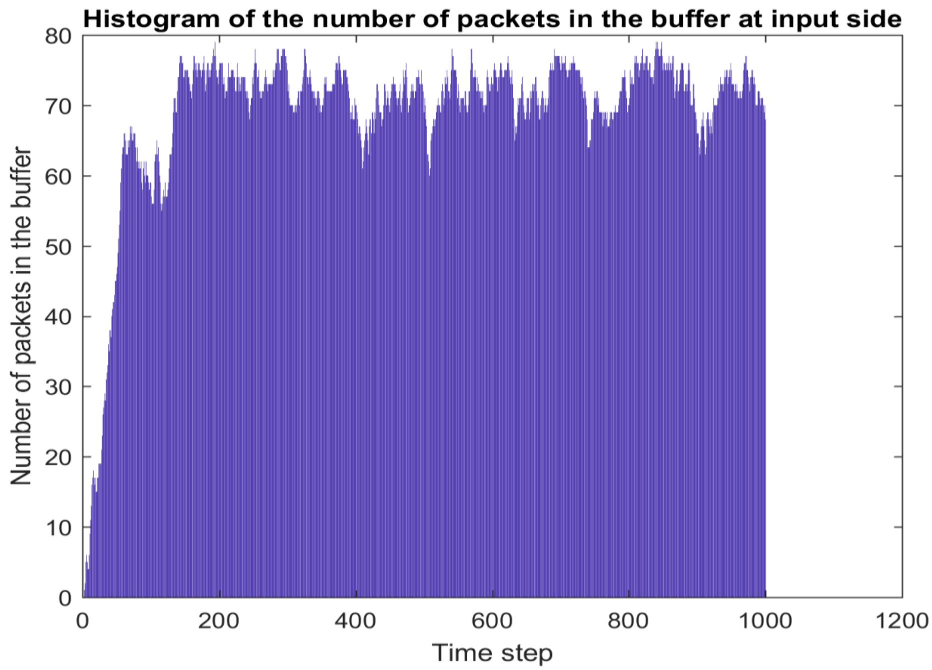
Balanced Traffic with probability of arrival of packets = 0.5 and Buffer Size = 10



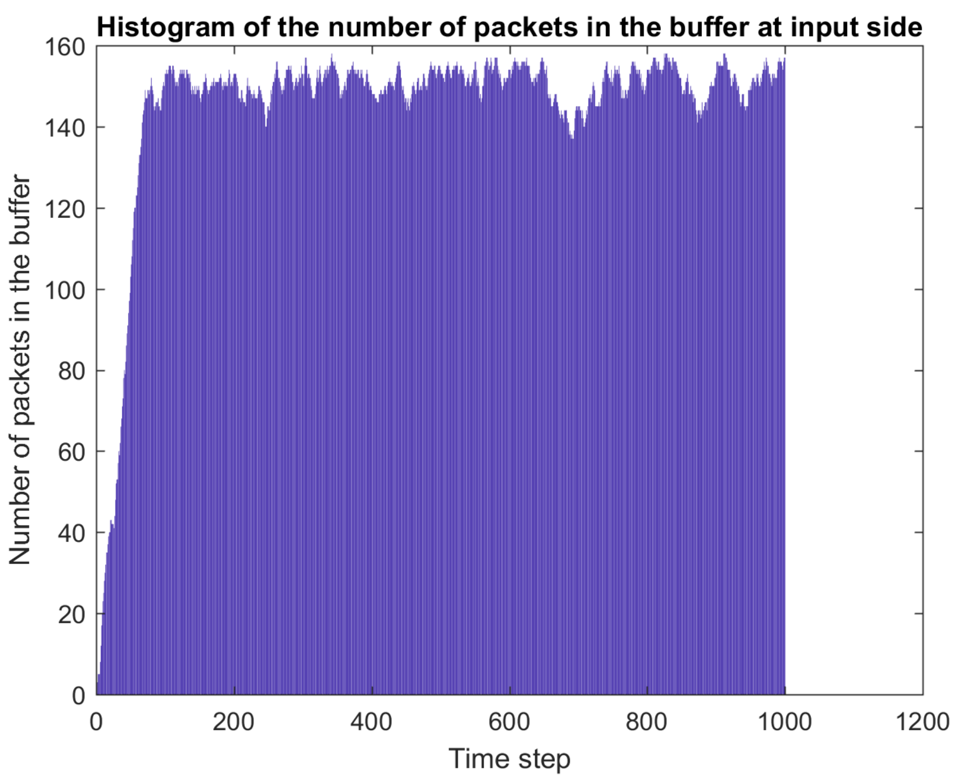
Balanced Traffic with probability of arrival of packets = 0.5 and Buffer Size = 20



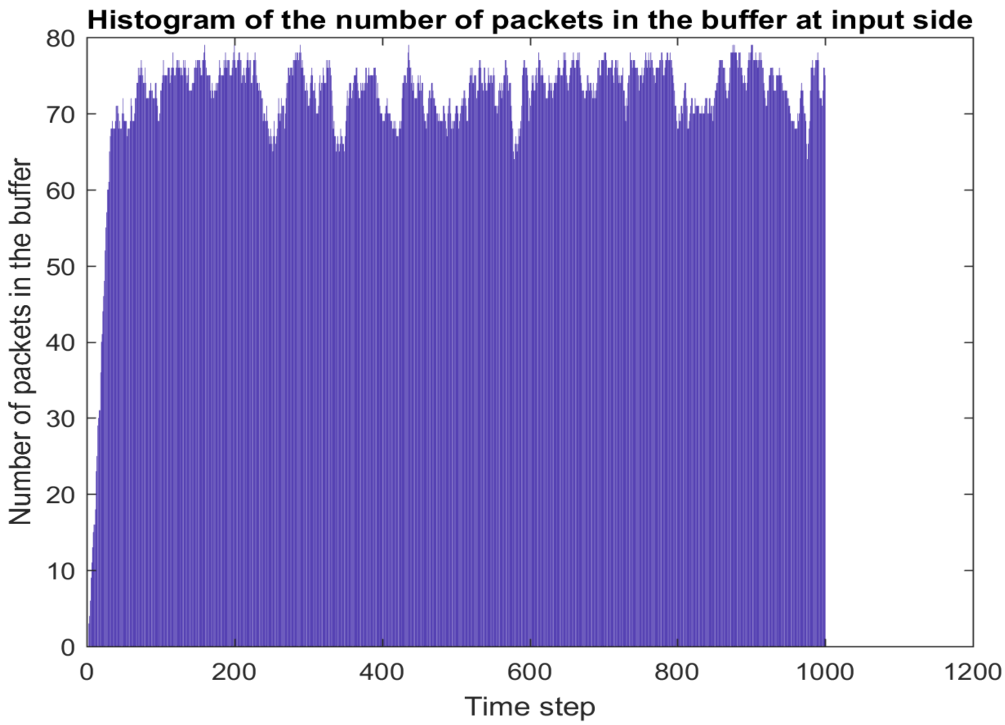
Balanced Traffic with probability of arrival of packets = 0.6 and Buffer Size = 10



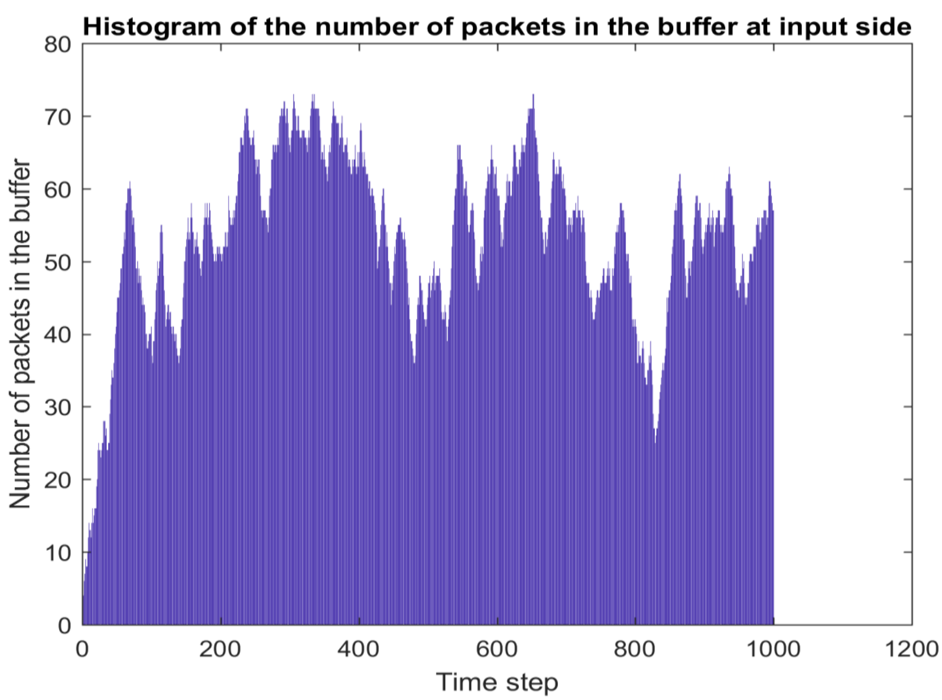
Balanced Traffic with k = 2, probability of arrival of packets = 0.5 and Buffer Size = 10



Balanced Traffic with k = 2, probability of arrival of packets = 0.5 and Buffer Size = 20



Balanced Traffic with k = 2, probability of arrival of packets = 0.6 and Buffer Size = 10



Balanced Traffic with k = 3, probability of arrival of packets = 0.5 and Buffer Size = 10

5. CODE

For Balanced Traffic,

```
close all;
clear;
clc;

prompt = 'What is the size of the switch?';
N = input(prompt)

prompt = 'What is the size of the buffer?';
B = input(prompt)

prompt = 'What is the arrival probability of a packet?';
Pa = input(prompt)

Buffer = zeros(N,B);
total = 0;
dropped = 0;
steps = 1000;
Ip = zeros(steps,1);
sum = 0;

for j = 1:steps

    Hash = zeros(N,1);

    for m = 1:N

        x = rand;
        if x < Pa
            z = find( Buffer(m,:) == 0)
            Z = isempty(z)
            if Z == 0
                Buffer(m,z(1)) = randi([1,N],1,1)
            else
                dropped = dropped + 1;
            end
        end
    end

    for m = 1:N
        if Buffer(m,1) == 0
            continue;
        end
        Hash( Buffer(m,1)) = Hash(Buffer(m,1)) + 1;
    end

    for m = 1:N
        if Hash(m,1) == 0
            continue;
        end
        Hash1(m,1) = randi([1,Hash(m,1)],1,1);
    end

    for m = 1:N
```

```
        if (Buffer(m,1) == 0)
            continue;
        end
        if (Hash1(Buffer(m,1)) == 0)
            continue;
        elseif (Hash1(Buffer(m,1)) == 1)
            Hash1(Buffer(m,1)) = 0;
            Buffer(m,:) = [Buffer(m,2:end),0]
            total = total + 1;
        else
            Hash1(Buffer(m,1)) = Hash1(Buffer(m,1)) - 1;
        end
    end

    Num = nnz(Buffer);
    Ip(j,1) = Num;
    sum = sum + Num;

end

figure;
bar(Ip);
title('Histogram of the number of packets in the buffer at input side');
xlabel('Time step');
ylabel('Number of packets in the buffer');
```

For Hot Spot Traffic,

```
close all;
clear;
clc;

prompt = 'What is the size of the switch?';
N = input(prompt)

prompt = 'What is the size of the buffer?';
B = input(prompt)

prompt = 'What is the arrival probability of a packet?';
Pa = input(prompt)

prompt = 'Enter value of k';
k = input(prompt)

Buffer = zeros(N,B);
total = 0;
dropped = 0;
steps = 1000;
Ip = zeros(steps,1);
sum = 0;

for j = 1:steps

    Hash = zeros(N,1);

    for m = 1:N
```

```
        x = rand;
        if x < Pa
            z = find( Buffer(m,:) == 0)
            Z = isempty(z)
            if Z == 0
                y = rand;
                if y < 1/k
                    Buffer(m,z(1)) = 1;
                else
                    Buffer(m,z(1)) = randi([2,N],1,1);
                end
            else
                dropped = dropped + 1;
            end
        end
    end

    for m = 1:N
        if Buffer(m,1) == 0
            continue;
        end
        Hash( Buffer(m,1)) = Hash(Buffer(m,1)) + 1;
    end

    for m = 1:N
        if Hash(m,1) == 0
            continue;
        end
        Hash1(m,1) = randi([1,Hash(m,1)],1,1);
    end

    for m = 1:N
        if (Buffer(m,1) == 0)
            continue;
        end
        if (Hash1(Buffer(m,1)) == 0)
            continue;
        elseif (Hash1(Buffer(m,1)) == 1)
            Hash1(Buffer(m,1)) = 0;
            Buffer(m,:) = [Buffer(m,2:end),0]
            total = total + 1;
        else
            Hash1(Buffer(m,1)) = Hash1(Buffer(m,1)) - 1;
        end
    end

    Num = nnz(Buffer);
    Ip(j,1) = Num;
    sum = sum + Num;

end

figure;
bar(Ip);
title('Histogram of the number of packets in the buffer at input side');
xlabel('Time step');
ylabel('Number of packets in the buffer');
```