Faculdade de Engenharia da
Universidade do Porto

Telecommunications Systems

Discrete event traffic simulation - Part 3

Daniela Faria,
Miguel Pires

21 de Maio de 2021

1 General characterization of the simulation methodology

For this exercise, we took the following constants:

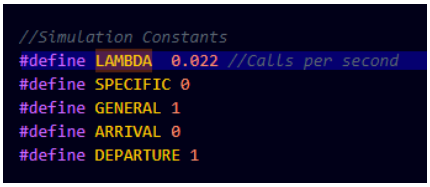

Figura 1: Constants

We also changed the structure that was provided into some more suitable to our needs.

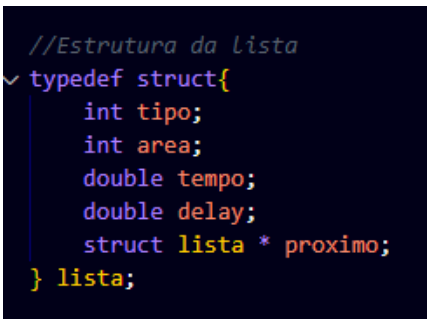

Figura 2: Struture

We built some auxiliary functions to ease the computation of the simulation that you can check in the func.h library.

2 Description of the simulation program

The first event is created. If it's an ARRIVAL type event, we determinate the following event in the same way we did before for the other parts. We check if there is any general purpose operator free. In case that one of them is free, we process it and generate a the DEPARTURE event. If there are not, we check if there is a space in the queue of the general purpose calls (which is defined by the input for the size of the finite queue). If there is space, you create the you add the event to the queue list and increment the counter of the calls that are delayed. If there is not space in the queue e loose that call.

In case of being a DEPARTURE type of event we decrement we check if the queue of the general purpose system is free. If it's not we take care of that event. We check the delay and update the histogram. We also analyse the error on that we delivered to the caller for the waiting time to be answered, and we update the prediction histogram with that error.

if the even is a SPECIFIC event, we change to the specific event list and we do almost the same procedure, but we don't have a finite queue anymore.

You can test the code using this commands

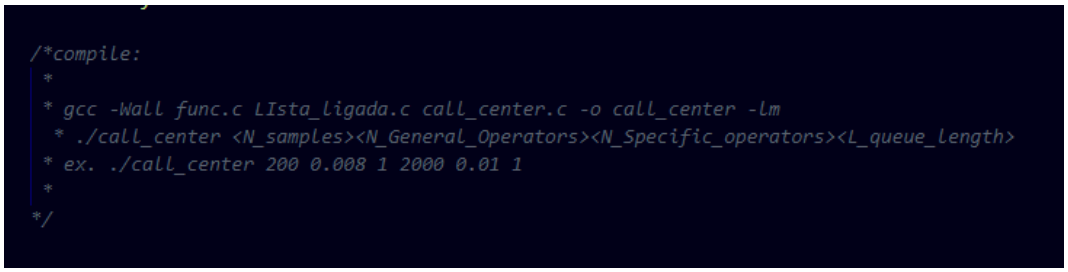

Figura 3: How to compile

3 Describe the algorithm used to predict the call waiting time

The waiting delay is calculated when there is an access to the waiting queue and an event from that list leaves. We used the formula provided by the professor:

$$avg[n] = avg[n-1] * (n-1)/n + current\_sample * 1/n$$

Figura 4: Running Average

4 Simulation Results

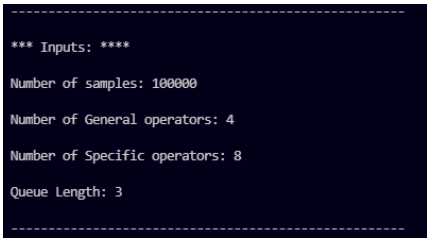

Figura 5: Inputs

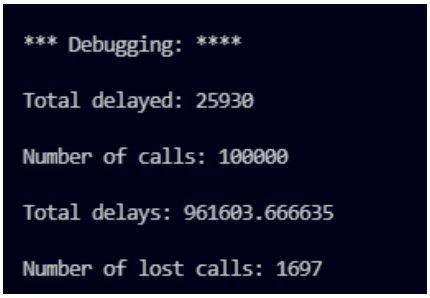

Figura 6: Criteria

In this program we identified some erros in the histogram output, we can say that the "histogram"criation is not in its best, which we tried to change and imporve with a creation of a new function for the histogram update.

```
Probability of a call being delayed at the input of the general purpose answering system: 25.930000 %
Probability of a call being lost at the input of the general purpose answering system: 1.697000 %
Average delay of calls for the calls that suffer delay at the input of the general purpose answering system: 37.084600 sec
Average of total delay of the calls since they arrive at the general purpose answering system until they answered by the are-specific answering system: 47.117176 sec
```

Figura 7: Criteria

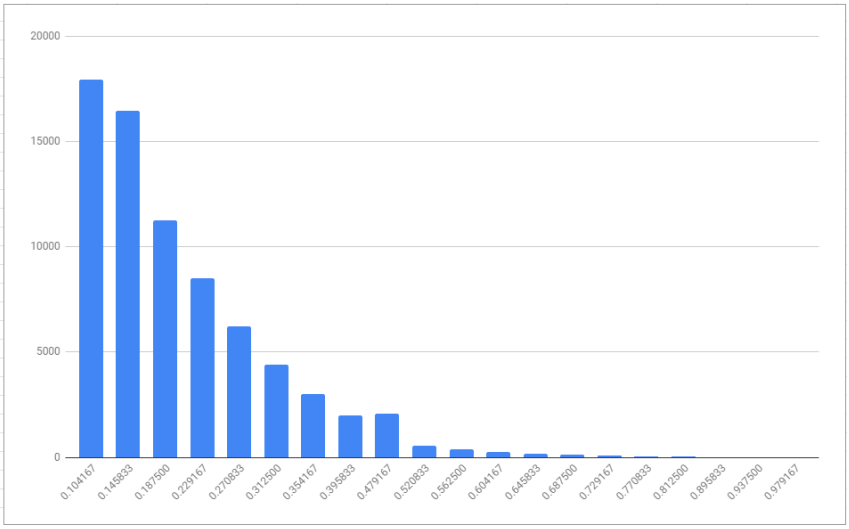

Figura 8: Delay Histogram

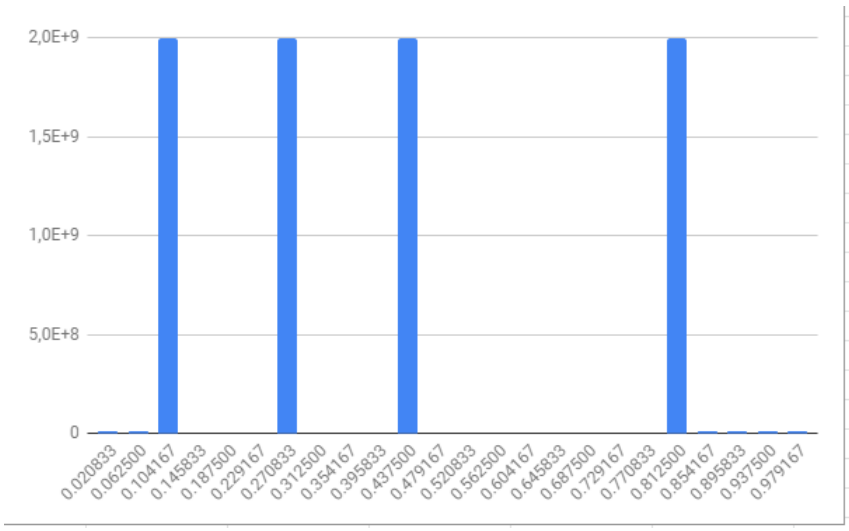

Figura 9: Prediction Error Histogram

5 Sensitivity Results

This part was not fulffiled as we expected. We saw that the results are now concident we the theory. We tried to see what was wrong, but we can see at least the growth of the values.

```
//desvio padrão

//Auxiliary variables
double aux=0, standard_devination = 0, standard_error_average= 0;
double confi_limite = 1.65;
double confi_interval = 0;


for(int j = 0; j<arrayCouter;j++){
    aux+=(arrayDelays[j]-array_average)*(arrayDelays[j]-array_average);
}

standard_devination = sqrt(aux / (arrayCouter-1));
standard_error_average = standard_devination / sqrt(arrayCouter);
confi_interval = confi_limite*standard_error_average;

//Print Parameters
```

Figura 10: Calculations

```
*** Sensitivity: ****

Sensitivity for lambda equal to 80 calls/hour

Absolute Prediction error: 22.499833

Relative Prediction error: 14.364055

Standard Devination: 22.682179

Standard Error Average: 0.086653

Confidance Interval, with limit of 90%: 0.142978
```

Figura 11: Prediction Error Histogram

Lambda	Ic
60	35.658978 +- 0.086586
80	46.595725 +- 0.139424
100	58.361549 +- 0.183679
120	73.631499 +- 0.226664

Figura 12: Ic

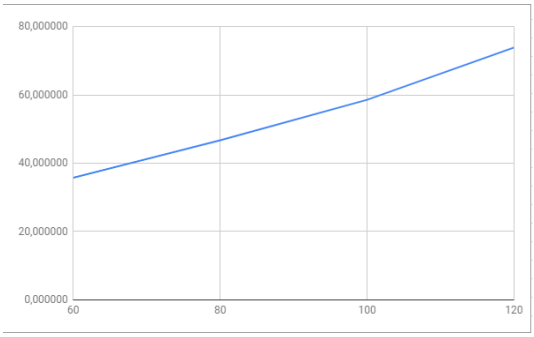

Figura 13: Ic

6 Conclusion

We can conclude that although the results were not the most desired ones, we achieved a conclusion of what is a emergency call system.