System Modeling and Simulation: Assignment 2 Report

Sina Molazemhosseini

October 5th, 2011

What follows is the summary of our work in this assignment 2. We have used Java with NetBeans IDE 6.9.1 to accomplish the tasks. It is also advisable to use this platform to run our simulation.

N.B. Attached to this document is the project folder containing the source files to run the simulation as well as two excel files that contains our data analysis as well as the confidence interval calculation. The aforementioned folder can be directly opened as a project by any version of NetBeans. Manually the source java files can also work with a simple command prompt mentioned in the assignment.

1 Problem Description and Model Development

1.1 Description of Model

Our model represents a Distribute Event Simulation of a mobile network for a highway with fixed length and a fixed number of base stations. Passing vehicles can start phone calls throughout the highway. Each call encapsulates the time of the call, the speed of the vehicle, the duration of the call and the position of the vehicle from the start of the highway. Each base station has a specific number of channels, some ordinary and some reserved that can be assigned to the cars nearest to them. Base stations cover the highway monotonously. They do not overlap and there's no area that's not being covered by a base station (where a base station's territory ends, the adjacent base stations territory begins).

Upon making a call, if there capacity of the base station is occupied with other calls, the call is blocked. If a passing cars moves from the territory of a base station to the territory of the next base station, then the call must also be transferred to the new base station. This event is called a handover. If an already established call moves to a new base station territory but is unable to find an empty channel on it, the call is considered as dropped.

What we are trying to do here is to simulate the problem and try to change our variables so that we have the minimum amount of blocked calls or missed calls.

1.2 Events

1. Arrival: indicates the initiation of a call;
2. Handover: indicates the handover of a call from one base station to another;
3. Departure: indicates the finishing of a call;

1.3 State Variables

Our state variables are as following:

1. Number of reserved channels;
2. Number of ordinary channels;
3. Timestamp of the model (Clock): The one which progresses as events occur;
4. FEL (Future Event List).

1.4 The Boundary Problem

We avoided the boundary problem by making the highway circular i.e. if a call is made just before the finishing line of the highway and is supposed to last after the finishing line it will be assigned to the starting base stations so that we have a proper load balancing on our calls.

2 Input Data Analysis

The attached MS Excel file (*data_ analysis.xlsx*) contains all the data analysis of our model based on the data provided to us. It contains 5 sheets: 4 calculating the distributions and one for the chi-square table. Each distribution contains all the necessary data together with a histogram that we use for the primary guess of the type of our distribution. None of our null-hypothesizes were rejected.

Our distributions turned out as following:

1. Inter-arrival times: An exponential distribution with a λ of 2.084;
2. Speed: A normal distribution with μ of 100.95525 and $\sigma^2$of 415.17 (standard devation of 20.375);
3. Position: A uniform distribution of with a b of 40.024;
4. Duration: An exponential distribution with a $\lambda$of 0.005.

3 Verification and Validation of the Model

3.1 Katharina's numbers

We first experimented with Katharina's numbers with the distributions used in the example for the sake of testing. Number of replications is 100. Here are the results.

```
Input:
Number of channels = 10
Number of reserved channels = 2
Replication length = 3600 seconds
Output:
Percentage of dropped calls: 0.49
Percentage of blocked calls: 0.18
---------------------------------------------------
Input:
Number of channels = 11
Number of reserved channels = 0
Replication length = 3600 seconds
Output:
Percentage of dropped calls: 0.18
Percentage of blocked calls: 0.04
```

The results were promising in comparison with the numbers of the example results.

3.2 Our own distributions

Next we tried out the derived numbers from our own calculated distributions. Here are the results:

```
Input:
Number of channels = 10
Number of reserved channels = 2
Replication length = 3600 seconds
Output:
Percentage of dropped calls: 2.2
Percentage of blocked calls: 1.29
---------------------------------------------------
Input:
Number of channels = 11
Number of reserved channels = 0
Replication length = 3600 seconds
Output:
Percentage of dropped calls: 1.11
Percentage of blocked calls: 0.47
```

In both occasions the numbers do not exceed the thresholds of 2.5 and 5.0 for respectively percentage of dropped calls and blocked calls.

Next we increased the replication rate from 3600 to 13600:

```
Input:
Number of channels = 10
Number of reserved channels = 2
Replication length = 13600 seconds
Output:
Percentage of dropped calls: 2.96
Percentage of blocked calls: 1.65
```

As can be seen, by increasing the replication length the percentages slightly increase. This is due to the fact that before the simulation is warmed up we might have less dropped and blocked calls.

Next we perform a stress test: we decrease the number of calls by reducing $\lambda$from 0.48 (our derived value) to 0.2:

```
Input:
Number of channels = 10
Number of reserved channels = 2
Replication length = 13600 seconds
Output:
Percentage of dropped calls: 0.02
Percentage of blocked calls: 0.01
Total calls average: 2708
```

Next test is on speed. Out speed had a normal distribution of μ of 100.95525 and $\sigma^2$of 415.17 (standard devation of 20.375).

```
Output:
Percentage of dropped calls: 2.96
Percentage of blocked calls: 1.65
```

By increasing the speed from 100 to 200 we'll have:

```
Output:
Percentage of dropped calls: 4.45
Percentage of blocked calls: 1.44
```

That means the number of blocked calls will double and it makes sense the cars pass the base stations quickly and we have more dropped calls.

Our duration had a exponential distribution of $\lambda$of 0.005. We decrease this number to 0.004. It will lead to an increase of calls duration to and:

```
Output:
Percentage of dropped calls: 7.14
Percentage of blocked calls: 3.45
```

4 Output Data Analysis

4.1 Warm-up

We experimented our system with various valuues for warm-up. The initial values so far were derived with a warm-up of zero seconds. Here are some other numbers:

```
Input:
Warm-up: 50
Output:
Percentage of dropped calls: 2.37
Percentage of blocked calls: 1.4
---------------------------------------------------
Input:
Warm-up: 100
Output:
Percentage of dropped calls: 2.41
Percentage of blocked calls: 1.6
---------------------------------------------------
Input:
Warm-up: 200
Output:
Percentage of dropped calls: 2.48
Percentage of blocked calls: 1.09
---------------------------------------------------
Input:
Warm-up: 300
Output:
Percentage of dropped calls: 2.51
Percentage of blocked calls: 1.11
---------------------------------------------------
Input:
Warm-up: 500
Output:
Percentage of dropped calls: 2.61
Percentage of blocked calls: 1.14
---------------------------------------------------
Input:
Warm-up: 1000
Output:
Percentage of dropped calls: 2.70
Percentage of blocked calls: 1.19
---------------------------------------------------
Input:
Warm-up: 2000
Output:
```

```
Percentage of dropped calls: 2.76
Percentage of blocked calls: 1.2
---------------------------------------------------
```

As can be seen from the result, for the case of percentage of blocked calls, we do not face a problem seeing it always remains below the threshold of 5.0. But starting from the warm-up value 200 to 300, the percentage of dropped calls passes the threshold of 2.5 that was expected from our system. But the QoS was instantly insured when we added the number of our channels by one. Now with 11 channels, the percentage of our dropped calls decreased back to normal. So if we want to have a better quality of service it would be of good practice if we increase the number of channels. Though as you can see, adding one channel only stablizes our results for a wide domain of replication time. Our system nonetheless performd well with a warm-up value of 200-300.

4.2 Confidence Interval

We calculated the confidence interval for our dropped calls and blocked calls in the second attached Excel file (Output data analysis). The results are in two different sheets:

The confidence arrival for our dropped calls is 39.89 ± 2.35

The confidence arrival for our blocked calls is 17.89 ± 1.08