CS5233 Assignment 2A

Lu Yicheng	A0268497E	Han Minghao	A0268533X
Wang Haowen	A0254309E	Wang Ziang	A0254362H

Problem Description and Model Development

SIMTel is a telecommunication company that operates its mob=ile telephony system along a 40-km highway. This paper employs a modeling approach to assess the existing system of SIMTel Corporation. By evaluating the quality of service, we provide a basis for determining whether to increase the subscriber and recommend feasible expansion proposals to SIMTel.

Assumption

It's a **one-way** highway with length of **40km**. Every **2km** a cell. Every cell has **10** channels. **Fixed** channel allocation is used. Handover time is **negligible**. **Hard handovers** are used.

State Variables and Events

- State Variables

  For a call

interval:	interval between two call arival
loc:	location of the call in the one-way highway (map with the **cell**)
call_duration:	duration of a call
speed:	moving speed of a call (cause handover)
cell:	current cell serving the call
channel:	current channel occupied by the call

  For System

channels:	available channels in each cell (10 channels × 20 cells)
cur_time:	current time
MoreTraffic -	expansion rate of subscriber

- Events

call_arrival:	current time + min(intervals), a new call arrives
get_channel:	the call tries to get one available channel from its cell
channel is None:	no available channel, the call is blocked
channel returned:	the call occupies this channel and successfully connected
handover:	the call moves to another cell and applies for handover
channel is None:	no channel available for handover, the call is dropped
channel returned:	successful handover, call occupies this channel and continues
dispose:	current time = end time of a call, the call is successful and ends

Description of Our Model

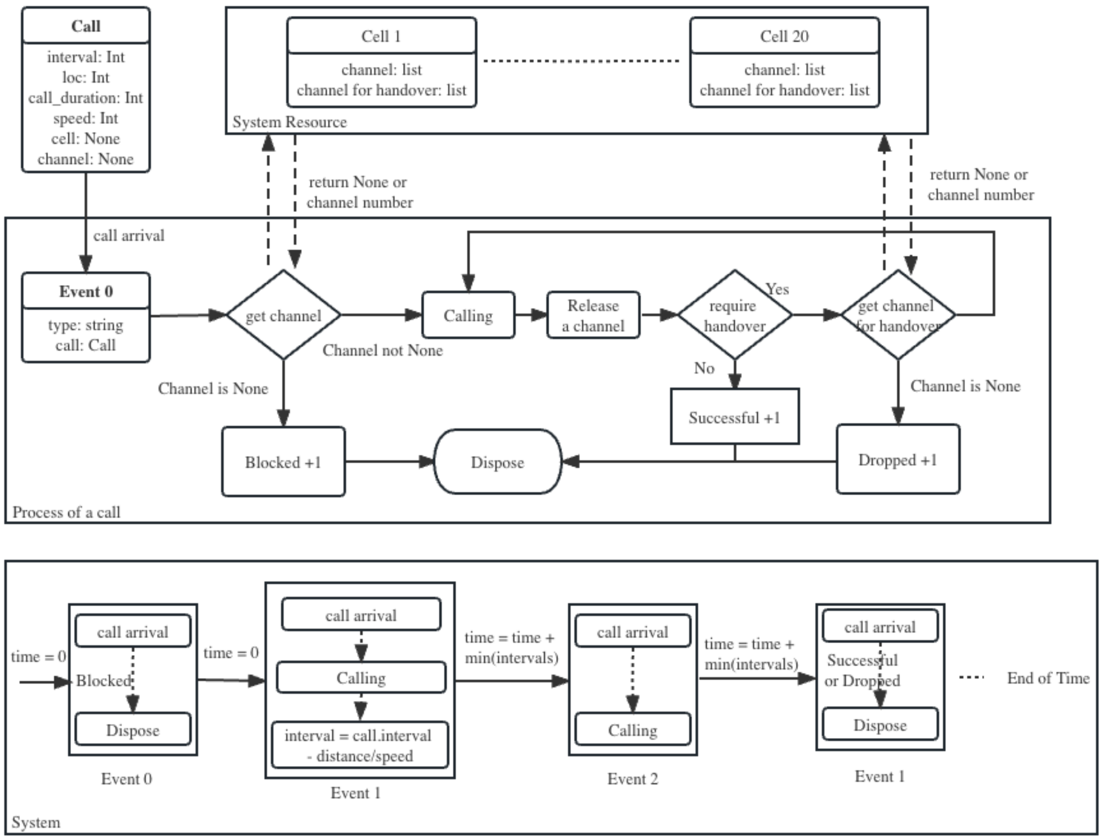


Figure1. our model simulating the SIMTel highway telephony system

- Data Structure

  Call : **Class**. Store the properties of a call, including interval, location, duration, speed, cell and channel (cell & channel are set as None, until one channel is allocated to this call).
  Cells : **List**. Contain 20 cells. Each is a dictionary with length initialized to 10.
  channel : **Dict**. Stored in Cells. The key is between 0 and 9. The value has a property of "handover", which means whether this channel is allocated for handover.
  event_list : **List**. Store all events. When push or pop an event from event_list, we use the python package of heapq to transform the **List** into **Min Heap** to ensure first processing the event which occurs earliest, that is with minimum interval.

- Procedure

1. call_arrival: ***Initialize*** an instance of class Call. Use python package of random to ***generate*** the attributes of the call according to their data distribution respectively. ***Encapsulate*** the call with {type, Call}, which is an **Event**. Type is the state of the event. ***Push*** the event into event_list and wait for system to process.
2. get_channel: When system starts to process an Event with the state of "**call_arrival**", system ***gets*** its serving station from the attribute of location and ***applies*** an available

channel from the station. If None, the call turns **blocked** and removed. If YES, the call ***occupies*** the channel.

3. calling: ***Simulate*** the duration of a call. The attribute of duration would be ***changed*** into $Duration = Duration - (LengthOfCell - loc)/speed$, which means the duration left when the call comes to the end of the current cell. If the left duration is negative, the call turns **successful** and an Event with the state of "**dispose**" would be created and pushed int event_list. If positive, an Event with the state of "**handover_required**" would be created and ***pushed*** into event_list.
4. handover: When system processes an Event with the state of "**handover_required**" which means the call has reached the end of current cell and needs a handover, system would release the channel occupied currently and applies an available channel from the next cell. If None, the call turns **blocked** and removed. If YES, the call ***occupies*** the channel.

- Run Setup

  The SIMTel telephony system is a 24/7 system. So we set **30** replications with each replication lasting **30×24×3600** seconds (720 hours) with warm-up time of **3600** seconds (1 hour) to simulate a full month's work. Base Time Units is set to **Seconds** to fit the call duration.

Input Data Analysis

Accoding to the raw data (callDuration, speed, call intervals, base station) provided by the professor, we converted the data into the format of txt and analyzed them with the input analyzer. We got the data distributions separately as follow:

- For callDuration

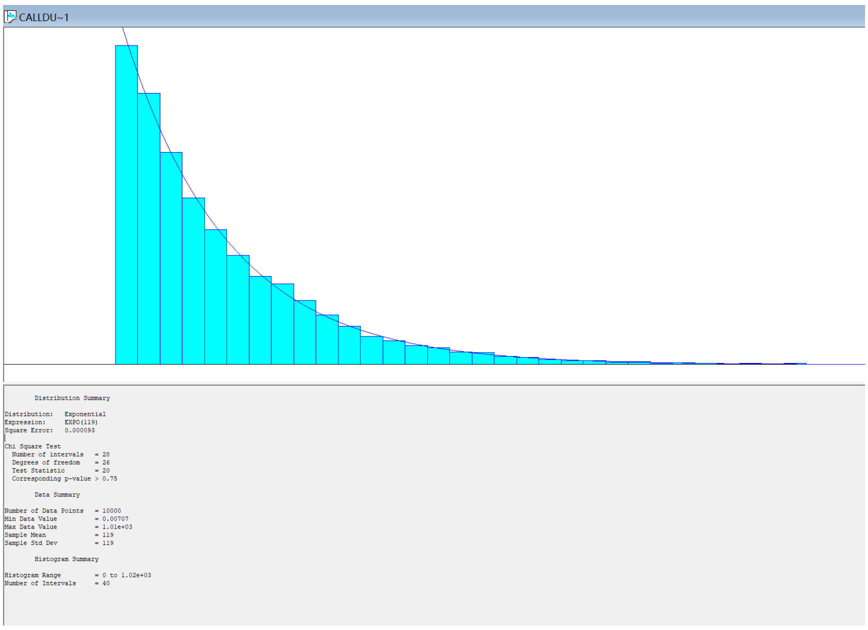


Figure2. data distribution of callDuration

In the picture of callDuration, it shows a typical **exponential distribution**, with the mean of 119. So we choose **exponential distribution** for callDuration.

- For speed

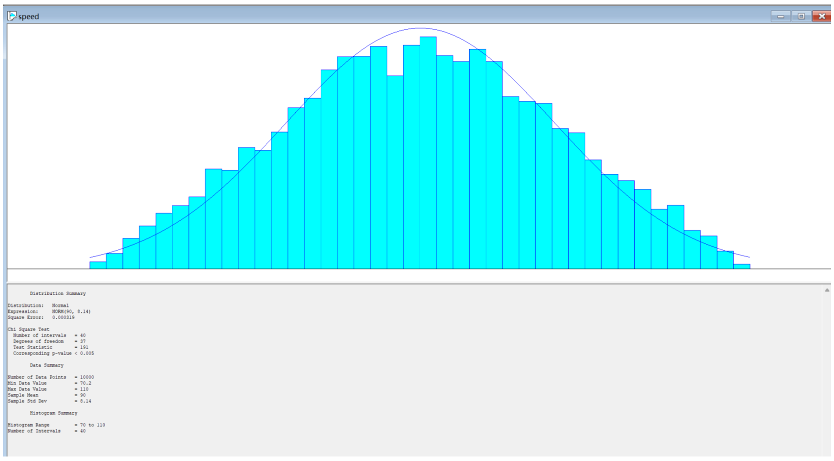
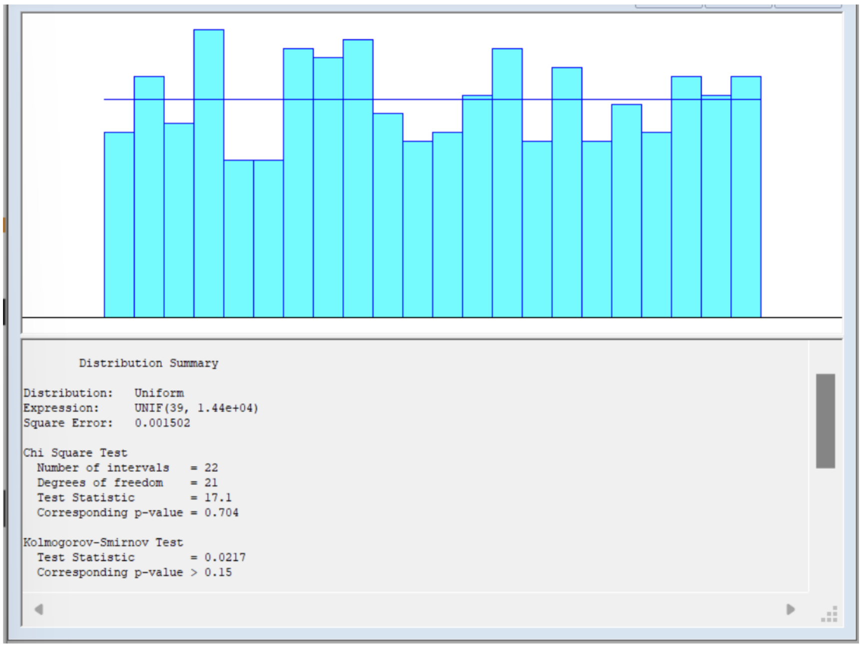


Figure3. data distribution of speed

In the picture of speed, it shows a **normal distribution**, with the mean of 90 and the standard deviation is 8.14. So we choose **normal distribution** for speed**.**

- For Call Interval



In the picture of Call Interval, it shows **uniform distribution**. So we choose **uniform distribution** for call interval.

- For Base Station

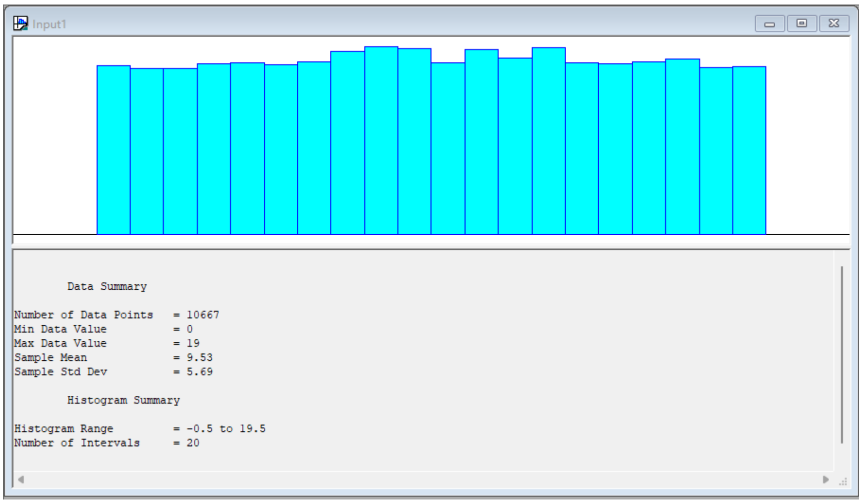


Figure4. data distribution of base station receiving calls

In the picture of base station, it shows **uniform distribution**. Every call has an even chance received by one station. So we choose **uniform distribution** for base station receiving calls.

Verification and Validation

Verification

Commonsense Methods

To verify the model's appropriate behaviors, the input parameters are adjusted to test whether the output exhibits the corresponding behavior.

When the number of channels for each cell is set to the default value of 10 channels per cell, the rate for blocked calls and dropped calls are 0.87% and 1.14% respectively. By increasing the number of channels to 12 per cell, the rate decreases to 0.15% and 0.21%, as expected. When the call duration is increased, both rates appear to be higher, at 2.30% and 0.47%. This is the expected behavior since longer call durations mean that calls will seize a channel for a longer time, leading to fewer available channels for other calls. Overall, the model demonstrates satisfactory behavior and generates expected outputs.

Structured Programming

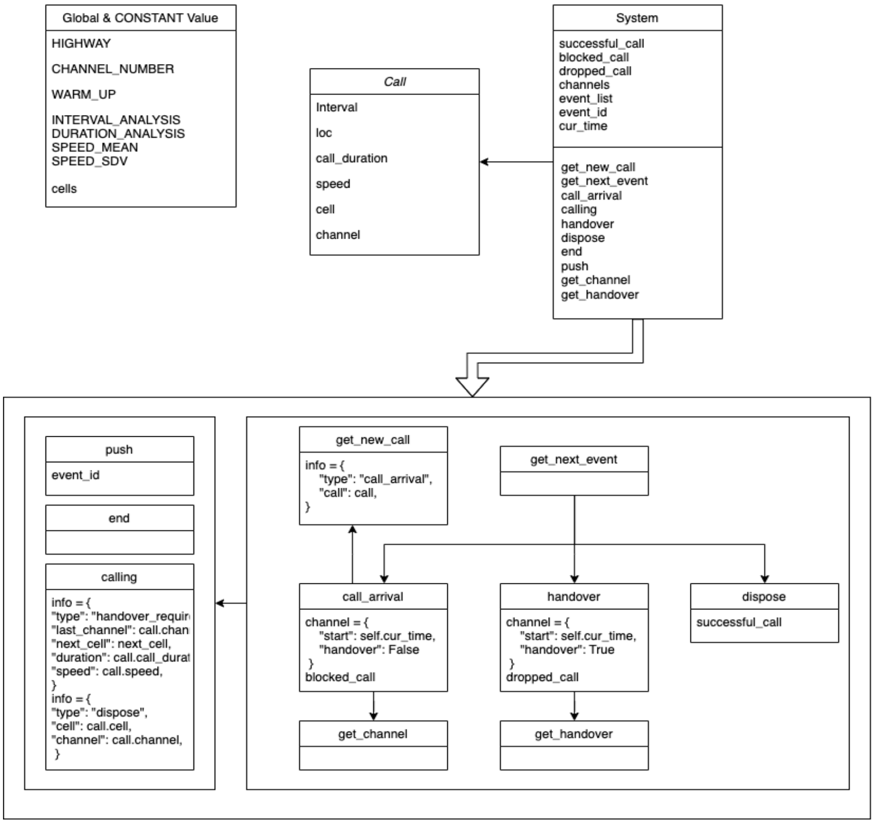


Figure5. structure of our program

The program code's structure is illustrated in the figure5:

The "System" class contains the main operating functions. It also accesses the "Call" class, which stores attributes for a call. There are three primary events for this model: call_arrival, indicating the creation of a call; handover, indicating a call being handed over to another cell; and dispose, indicating the end of a call. The code's structure aligns with the models represented in Figure 1.

Tracing the Simulation

The simulation's trace is recorded with the logging information in 'simulation.log'. It monitors the start of the simulation for each replication, the creation of new calls, the blocked and dropped call events, the disposal of calls, and the end of the simulation.

Program Testing

The program is tested with file ‘verfication.py’. The unit test includes the modification of several variables, such as speed, call duration, etc. It also tests if variables work properly.

By incorporating commonsense methods, structured programming, tracing the simulation, and program testing, the model has been thoroughly verified to ensure that it functions as intended and produces accurate and reliable results.

Validation

Output Data Analysis

Experiment 1

The objective of this experiment is to establish whether the **present** system is experiencing unsatisfactory call drops or blocks, which users have reported as a problem.

We set **30** replications with each replication lasting **30×24×3600** seconds (720 hours) with warm-up time of **3600** seconds (1 hour) to simulate a full month's work. Base Time Units is set to **Seconds** to fit the call duration.

The following table shows the simulation results of this experiment:

Run	Total Calls	Blocked Calls	Dropped Calls
1	1916675	12521	16933
2	1915879	12674	16833
3	1919382	12983	16841
4	1918723	12844	16915
5	1918151	12497	16856
6	1917205	12564	16842
7	1917574	12794	16657
8	1916476	12637	16666
9	1914871	12517	16905
10	1916537	12637	16724
11	1918275	13010	17127
12	1917220	12922	16999
13	1920229	13166	17152
14	1915439	12893	16698
15	1917572	12927	16780
16	1915568	12777	17036
17	1918501	12814	16921
18	1919377	12911	17011
19	1919415	12904	16929

20	1917956	12866	17057
21	1914576	12709	16703
22	1921150	12572	16791
23	1915596	12963	16730
24	1916570	12790	16863
25	1915343	12602	16572
26	1919104	12845	16767
27	1917899	12993	16906
28	1917335	12826	17319
29	1918997	12774	16836
30	1917503	12506	16807

We can compute the following statistics from above results:

- Overall point estimator of the percentage of blocked calls = 0.67%
- Standard error of blocking rate = 0.00001615
- 95% confidence interval of the percentage of blocked calls = 0.67% +/- 0.0027%
- Overall point estimator of the percentage of dropped calls = 0.88%
- Standard error of dropping rate = 0.00001484
- 95% confidence interval of the percentage of dropped calls = 0.88% +/- 0.0025%

It can be seen from the above results that the blocking rate is significantly less than 2%.

Apply hypothesis test on the dropping rate(with level of significance α = 0.05):

H_0: the percentage of dropped calls ≥ 1%

H_1: the percentage of dropped calls < 1%

$t_0 = \frac{0.0088 - 0.01}{0.00001484}$ = -80.86 < -$t_{29,\,0.05}$ = -1.70

Thus H_0 is rejected.

From the experimental results, it can be seen that both the blocking rate and dropping rate meet the requirements (2% and 1% respectively), **therefore, the current system can satisfy the users' needs.**

Experiment 2

The objective of this experiment is to explore whether the system can **handle more traffic with the same configuration** as that in experiment 1, except that the traffic is increased by 5%(call arrive interval is 0.95 * interval).
We set **30** replications with each replication lasting **30×24×3600** seconds (720 hours) with warm-up time of **3600** seconds (1 hour) to simulate a full month's work. Base Time Units is set to **Seconds** to fit the call duration.
The following table shows the simulation results of this experiment:

Run	Total Calls	Blocked Calls	Dropped Calls
1	2016750	17830	22996
2	2018069	17453	23097
3	2018454	17788	23259
4	2018172	17334	22611
5	2017671	17533	23207
6	2017771	17702	23037
7	2017734	17478	22905
8	2018067	17400	23080
9	2016463	17600	22750
10	2017442	17704	22770
11	2014734	17231	22648
12	2019513	17790	23017
13	2015461	17387	23160
14	2018376	17845	23136
15	2016455	17271	22651
16	2017700	17474	22976
17	2017315	17480	23272
18	2017098	17454	22758
19	2018695	17773	23449
20	2018607	17421	22994
21	2017270	17706	23099
22	2018398	17688	22995

23	2018539	17504	22854
24	2019378	17832	23324
25	2018941	17769	22864
26	2018539	17609	22983
27	2018928	17674	23233
28	2017182	17457	22889
29	2018007	17669	23217
30	2019905	17901	23173

We can compute the following statistics from above results:

- Overall point estimator of the percentage of blocked calls = 0.87%
- Standard error of blocking rate = 0.00001592
- 95% confidence interval of the percentage of blocked calls = 0.87% +/- 0.0027%
- Overall point estimator of the percentage of dropped calls = 1.14%
- Standard error of dropping rate = 0.00001851
- 95% confidence interval of the percentage of dropped calls = 1.14% +/- 0.0031%

It can be seen from the above results that the percentage of dropped calls exceeds the limit of 1%. **Therefore, without any changes to the configuration, the current system can not handle a 5% increase in traffic.**

Experiment 3

In Experiment 2, although the drop rate exceeded 1%, the block rate was well below the limit of 2%. So we can use this to explore whether it is possible to make the system meet the usage requirements by **adjusting the number of reserved channels for handovers**, which should reduce the dropping rate while increasing the blocking rate.

The objective of experiment 3 is to establish whether the present system can handle a 5% increase in traffic, when each one of the 20 cells has 1 channel reserved for handovers.

We set **30** replications with each replication lasting **30×24×3600** seconds (720 hours) with warm-up time of **3600** seconds (1 hour) to simulate a full month's work. Base Time Units is set to **Seconds** to fit the call duration.

The following table shows the simulation results of this experiment:

Run	Total Calls	Blocked Calls	Dropped Calls
1	2018487	39113	14892

2	2021763	39788	15029
3	2016971	39495	14720
4	2018099	39385	14451
5	2019198	39428	14913
6	2016774	39155	14672
7	2018807	39378	14851
8	2018060	38757	14839
9	2020756	39417	15030
10	2017606	39071	14679
11	2017647	38871	14730
12	2015078	39175	14623
13	2018278	38981	14544
14	2018330	39399	14967
15	2018534	39363	14868
16	2016633	38850	14857
17	2020021	39564	14965
18	2015814	38961	14515
19	2018549	39211	14824
20	2017626	38985	14617
21	2018750	39694	14968
22	2017373	39487	14515
23	2017715	39137	14592
24	2017890	39030	14705
25	2018803	39268	14895
26	2017805	39230	14752
27	2019208	39907	15303
28	2018248	39624	14929
29	2019508	39061	14858
30	2019846	39262	14855

We can compute the following statistics from above results:

- Overall point estimator of the percentage of blocked calls = 1.95%
- Standard error of blocking rate = 0.00002410
- 95% confidence interval of the percentage of blocked calls = 1.95% +/- 0.0041%
- Overall point estimator of the percentage of dropped calls = 0.73%
- Standard error of dropping rate = 0.00001611
- 95% confidence interval of the percentage of dropped calls = 0.73% +/- 0.0027%

It can be seen from the above results that the dropping rate is significantly less than 1%.

Apply hypothesis test on the blocking rate(with level of significance $\alpha = 0.05$):

H_0: the percentage of blocked calls ≥ 2%

H_1: the percentage of blocked calls < 2%

$t_0 = \frac{0.0195 - 0.02}{0.00002410}$ = -20.75 < $-t_{29,\,0.05}$ = -1.70

Thus H_0 is rejected.

From the experimental results, it can be seen that both the blocking rate and dropping rate meet the requirements (2% and 1% respectively), **therefore, the system can satisfy the users' needs.**

Conclusions

(1) Regarding the received complaints about interrupted calls, we evaluate the Quality of Service of SIMTel system with the model and data, and conclude that the current system **does not have a problem** regarding the quality of the service requierments.

(2) Regarding the above situation, we examined the performance of the current system with **a 5% increase in traffic**. The simulation result shows that the current system can not handle such an increase since the dropping rate exceeds the limit of 1% in the experiment.

(3) **Having 1 channel reserved for handovers** for each one of the 20 cells, the system can meet the service requierments with a 5% increase in traffic. However, the experiment shows that the system may not be able to handle more than 5% increase in traffic without adding more cells or increase the number of channels in each cell, since the upper bound of 95% confidence interval of blocking rate(which is 1.9541%) is fairly close to the limit of 2%.