Project 1 - Markovian Queue Implementation

Harshavardhan Nalajala

October 23 2017

Contents

1 Data Structures

- Min Heap to store the requests ordered by time. Event that occurs in the most near future is stored at the root of the heap. Max heap size is dependent on the requests generated. In the current project, heap size is dependent on L(no. of terminals) since each terminal cannot generate another request until current request from the terminal is processed by the system.
- Queue to store the requests that arrived at the system but servers are busy. Max queue size is order of K(Queue size).

2 ReadMe

- main.c contains the main simulation program (run_simulation API).

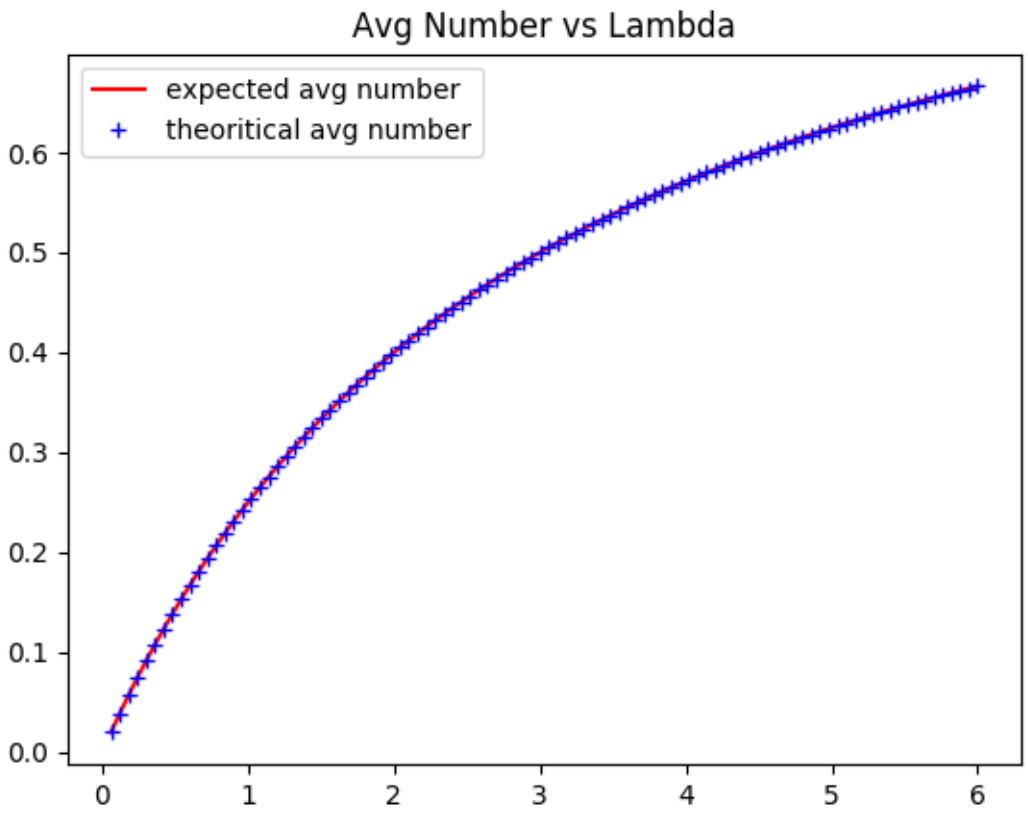
- utils.c is used to store data structures and generate exponential random variables.
- utils.h contains the structures and API used by main.c and implemented in utils.c
- input.in contains the input to be given to the program. Following is the input style.
  - no. of events to be generated for each run, 'run'
  - no. of terminals, L
  - queue size, K
  - no. of servers, m
  - service time of each server, 'mu'
- plotter.py python script to generate the graphs of output obtained from simulation.
- avg_output file contains the output as lambda, expected no., theoritical expected no., expected time, theoritical expected time, blocking probability, theoritical blocking probability, utilization of the system, theoritical utilization of the system.
- q_simulator executable to be run.
- run_script shell script file to build and run the simulation. Use following method to run the simulation.
  - modify input.in as per above instructions
  - ./run_script.sh

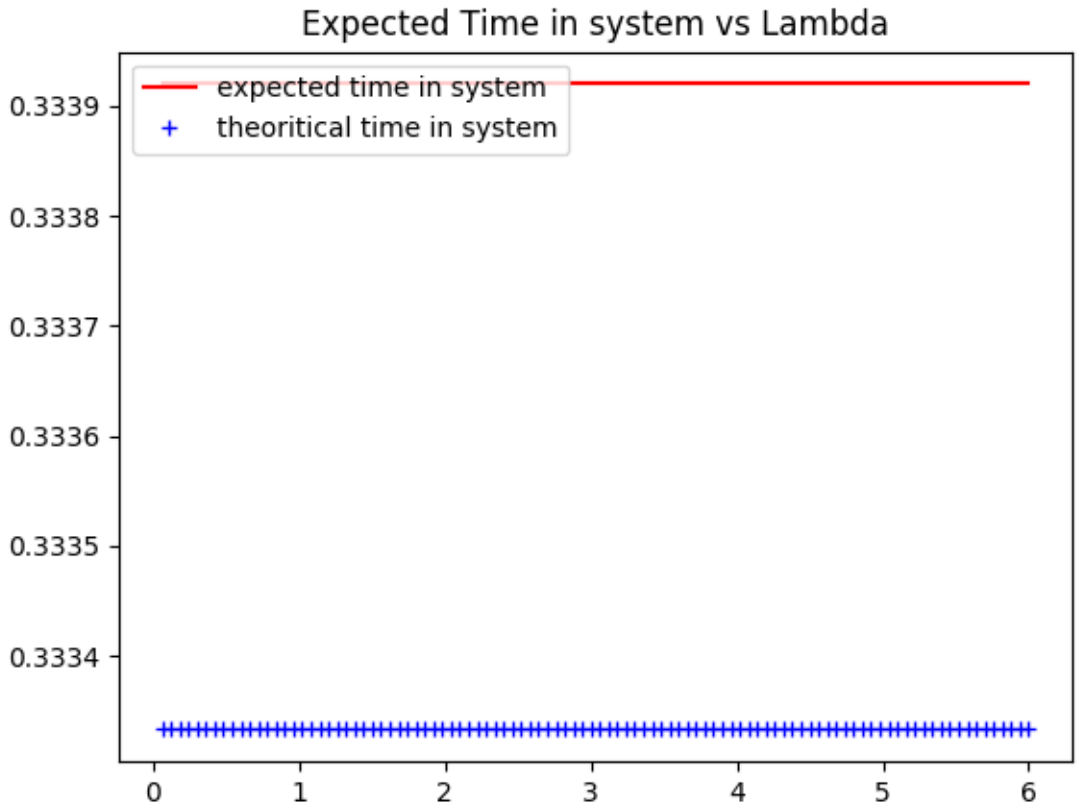
3 Dependencies

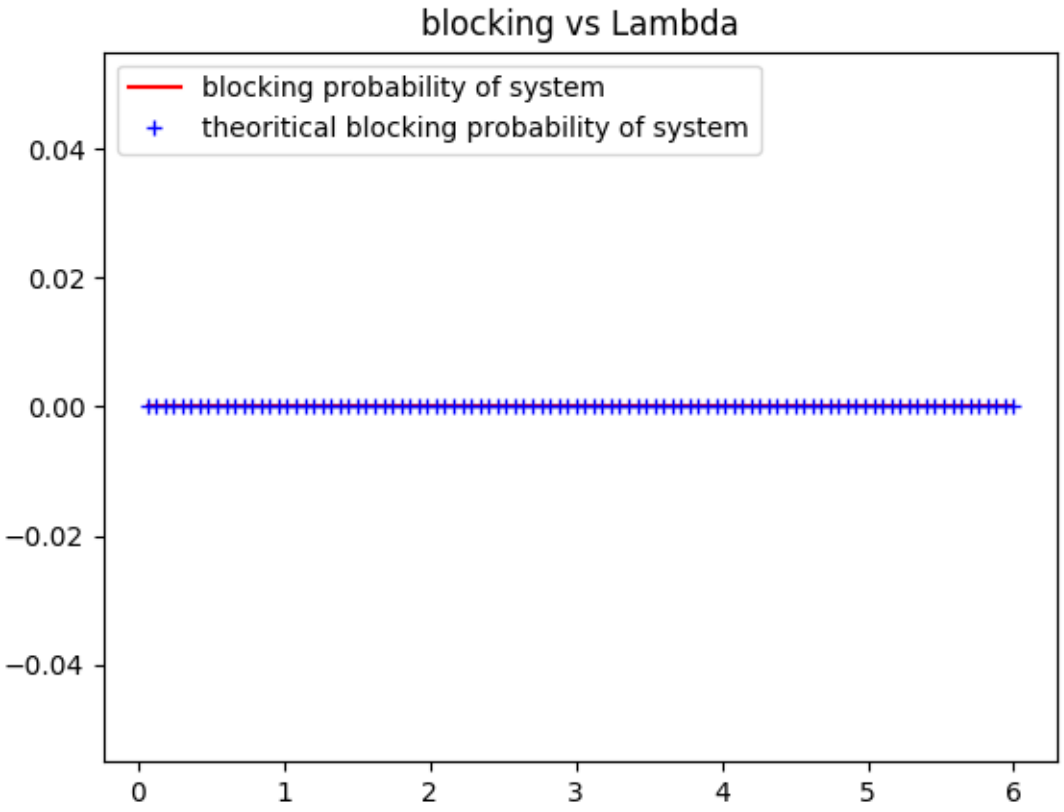
- gcc compiler
- python3. For python2 or 2.7, run_script needs to be modified to enable python2.

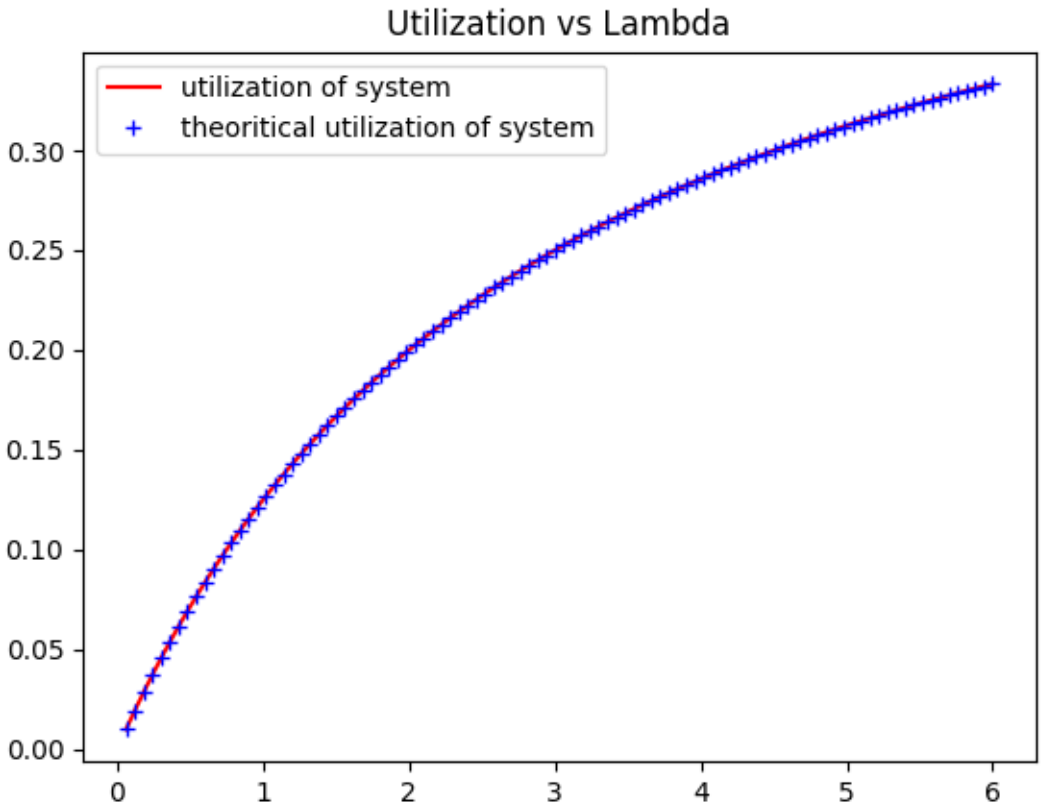
4 OUTPUTS

4.1 L1, K4, M2, U3

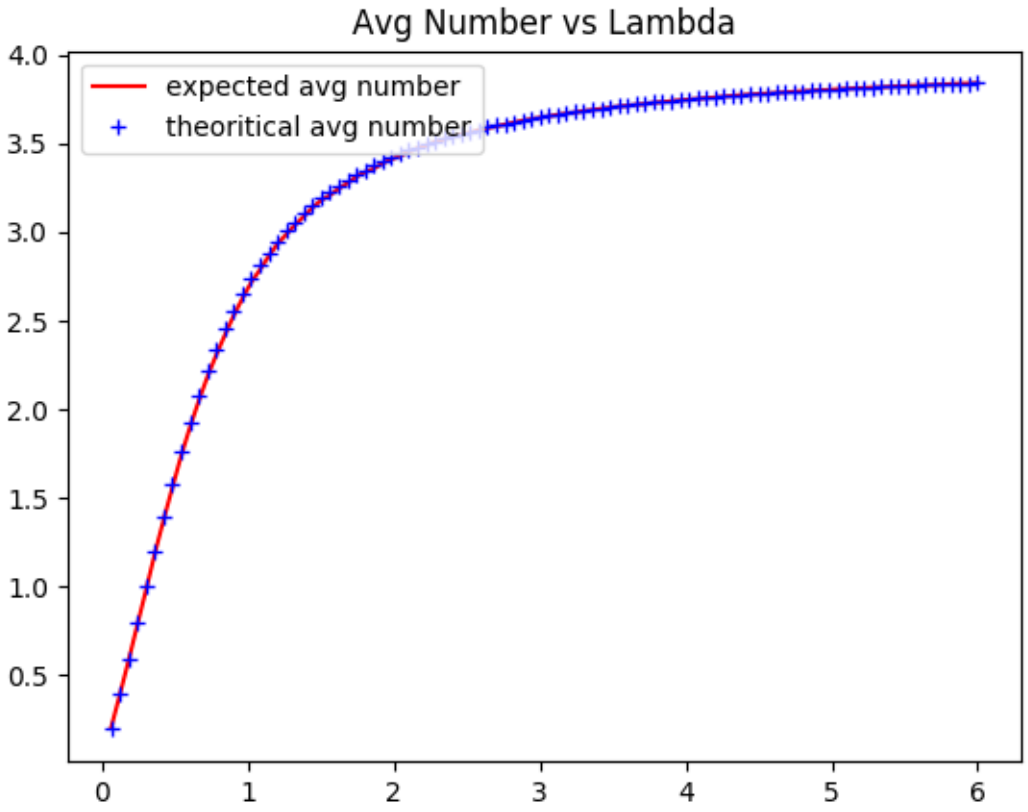


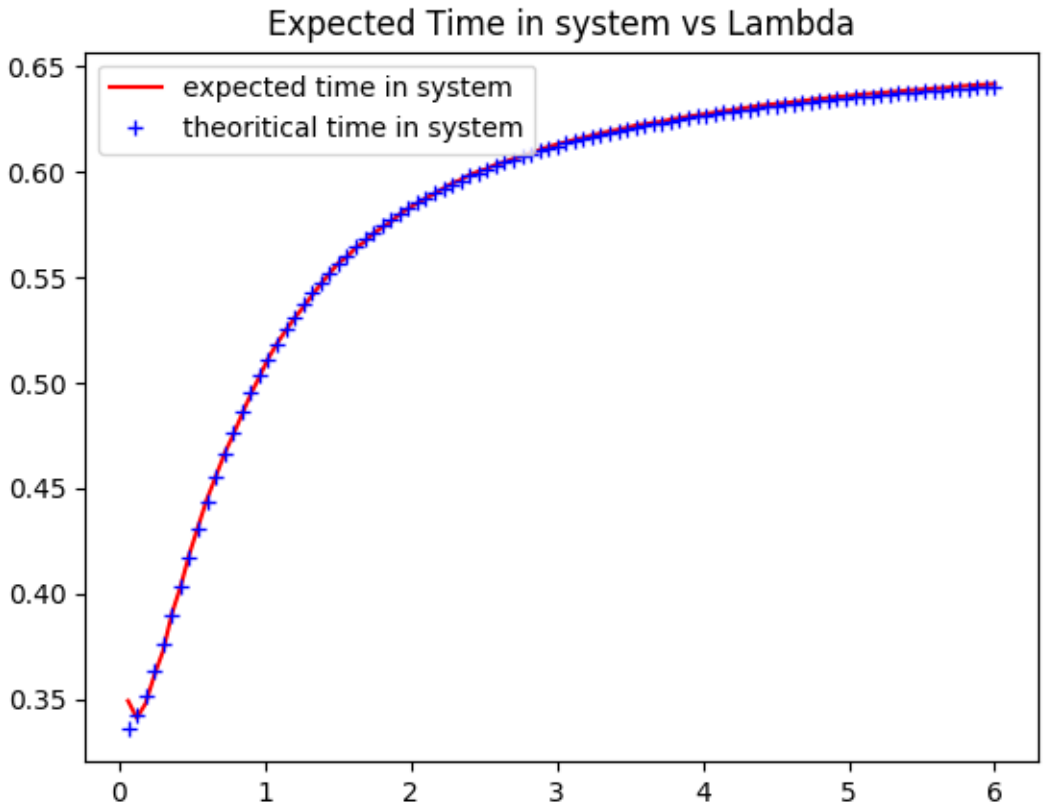

Expected Time in system vs Lambda
expected time in system
theoritical time in system
0.3339
0.3338
0.3337
0.3336
0.3335
0.3334
0
1
2
3
4
5
6

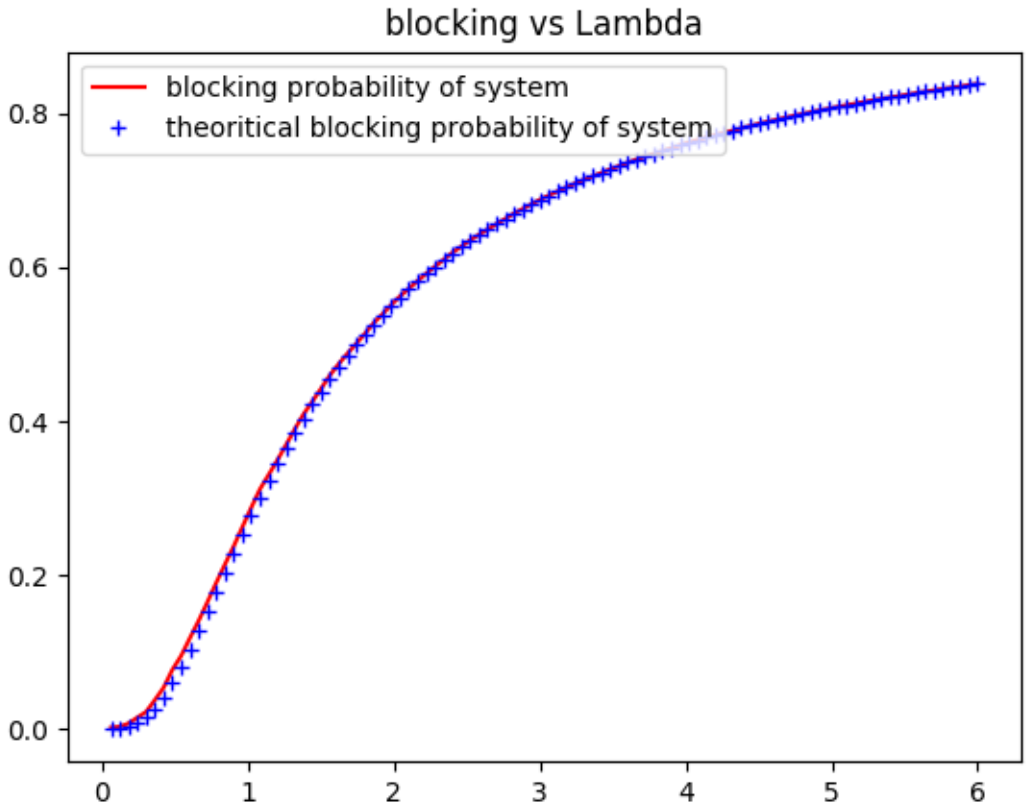

blocking vs Lambda
blocking probability of system
theoritical blocking probability of system
0.04
0.02
0.00
−0.02
−0.04
0
1
2
3
4
5
6

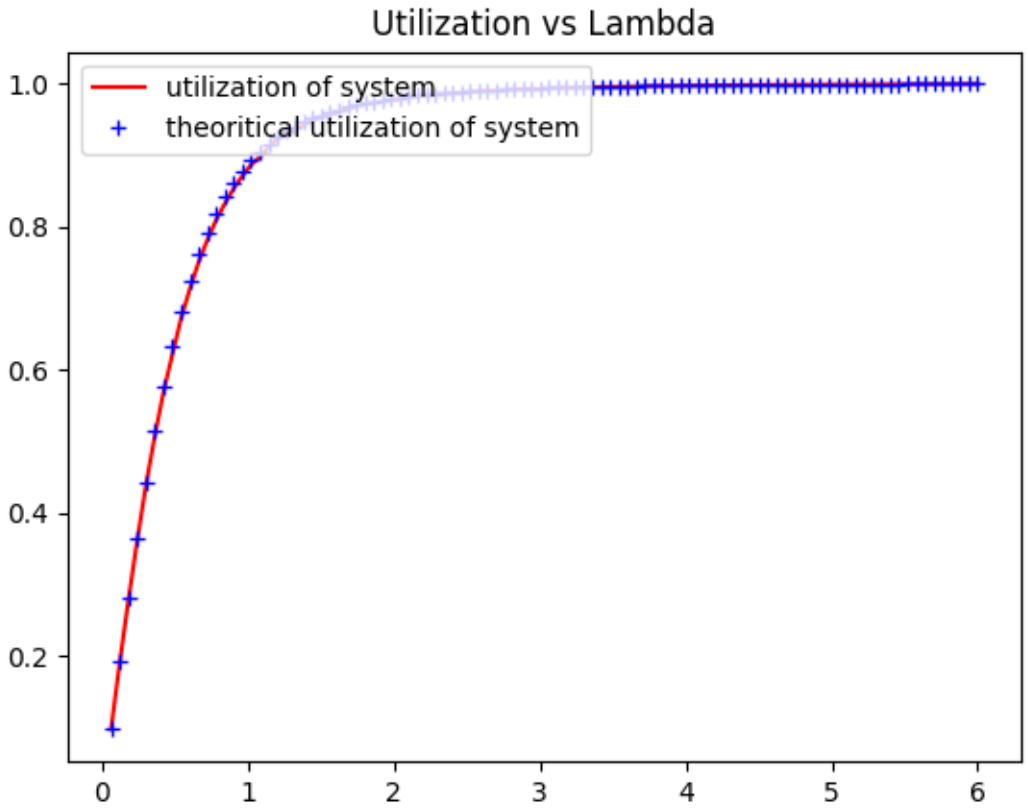

Utilization vs Lambda
utilization of system
theoritical utilization of system
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
1
2
3
4
5
6

4.2 L10, K4, M2, U3

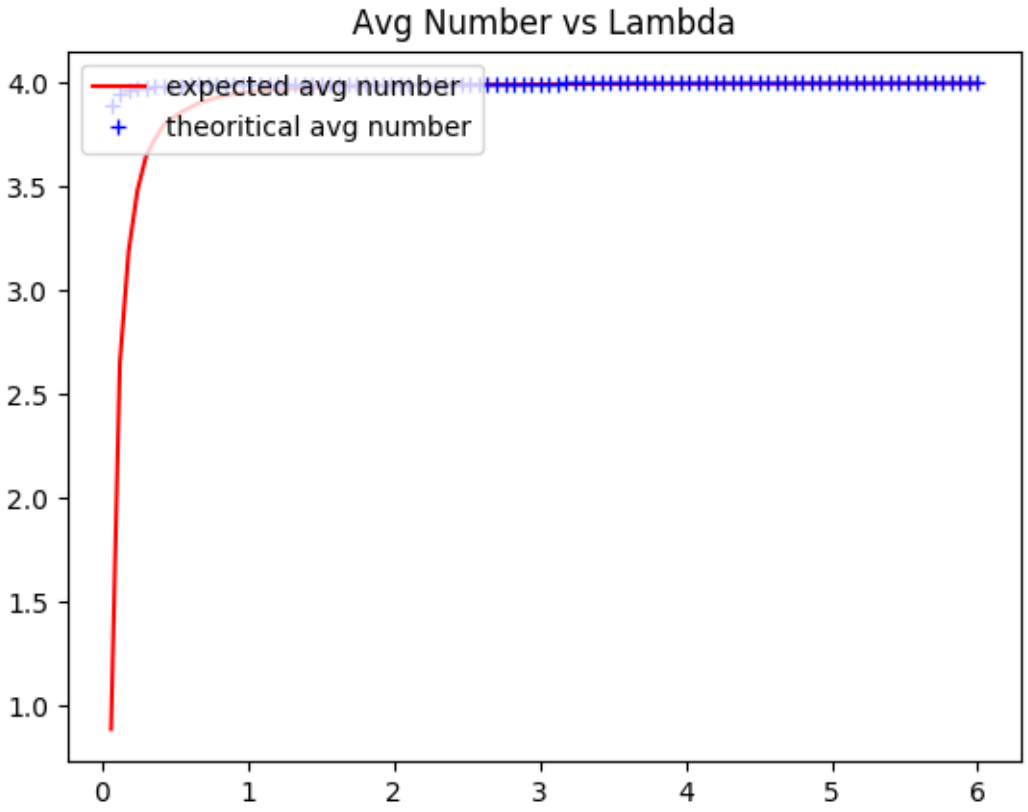


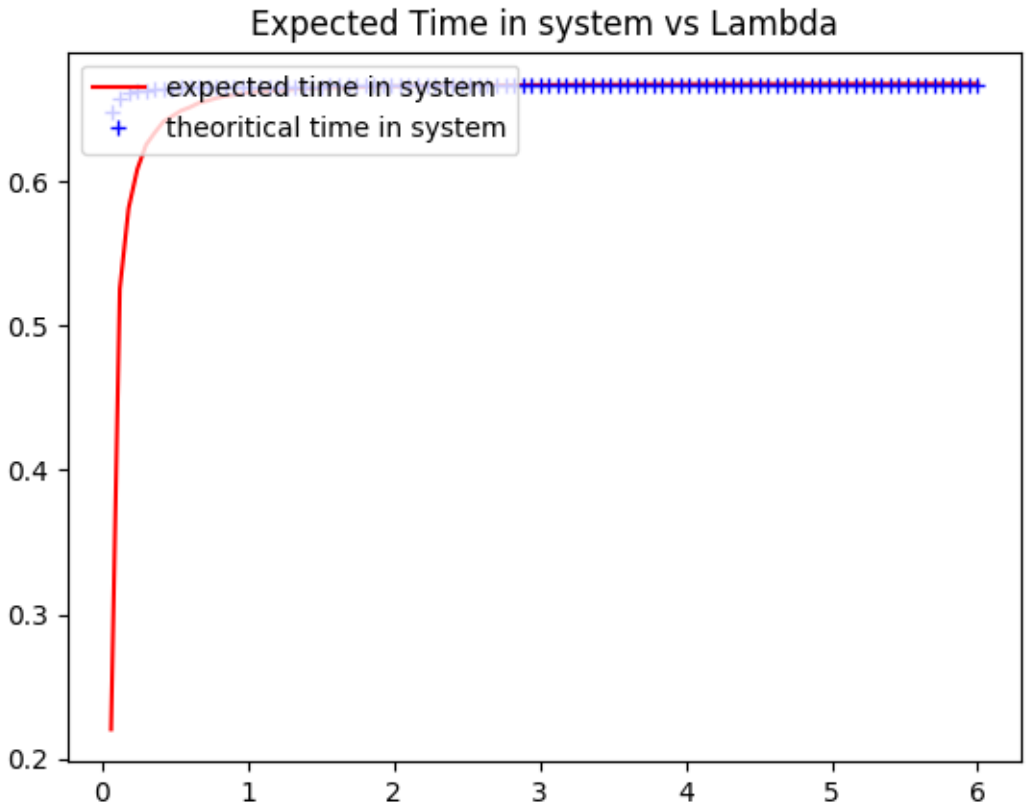

Expected Time in system vs Lambda
expected time in system
theoritical time in system
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0
1
2
3
4
5
6

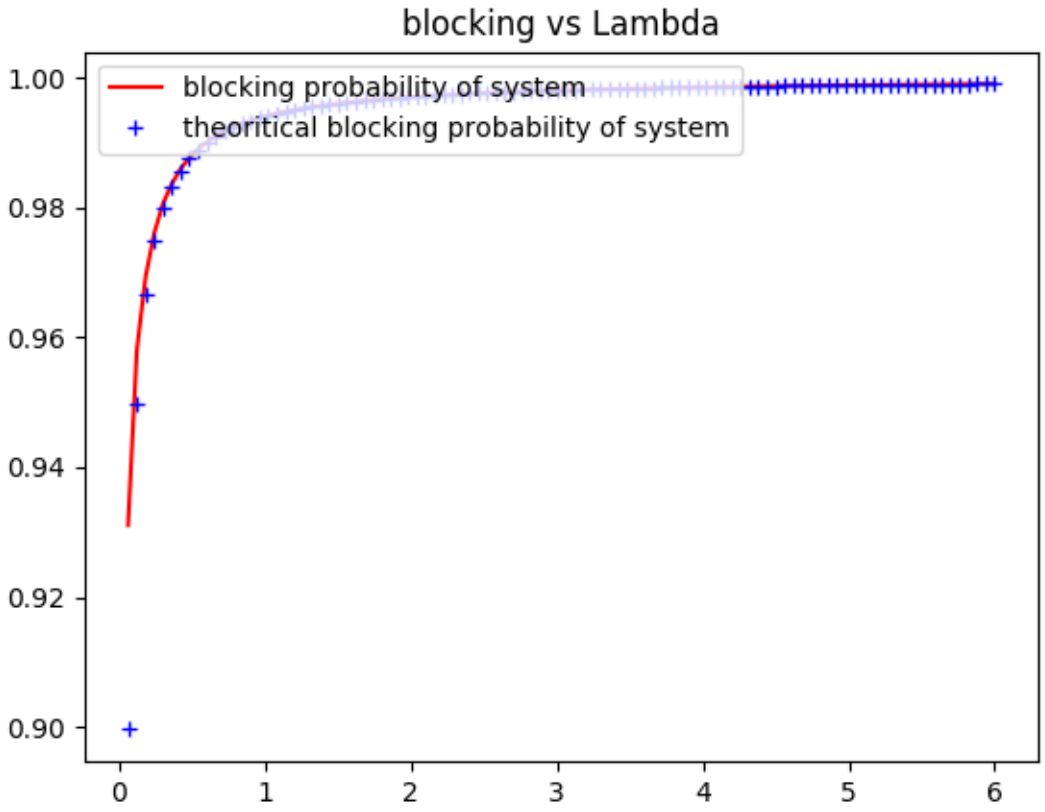

blocking vs Lambda
blocking probability of system
theoritical blocking probability of system
0.0
0.2
0.4
0.6
0.8
0
1
2
3
4
5
6

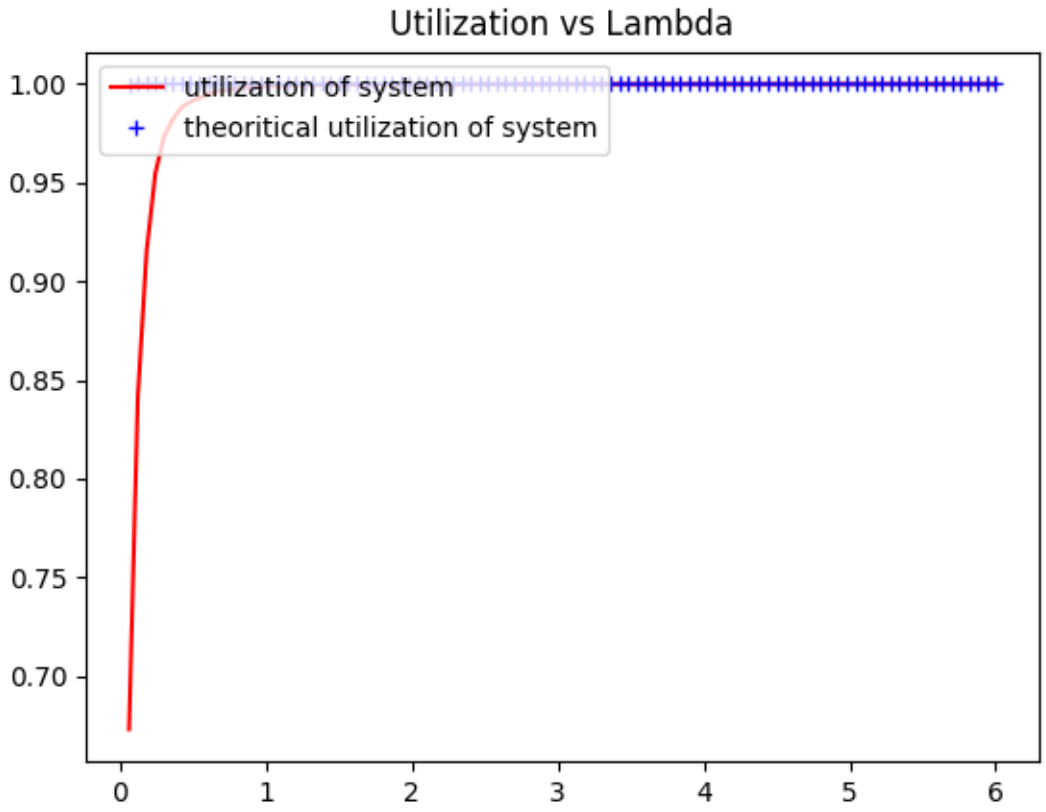

Utilization vs Lambda
utilization of system
theoritical utilization of system
1.0
0.8
0.6
0.4
0.2
0
1
2
3
4
5
6

4.3 L1000, K4, M2, U3

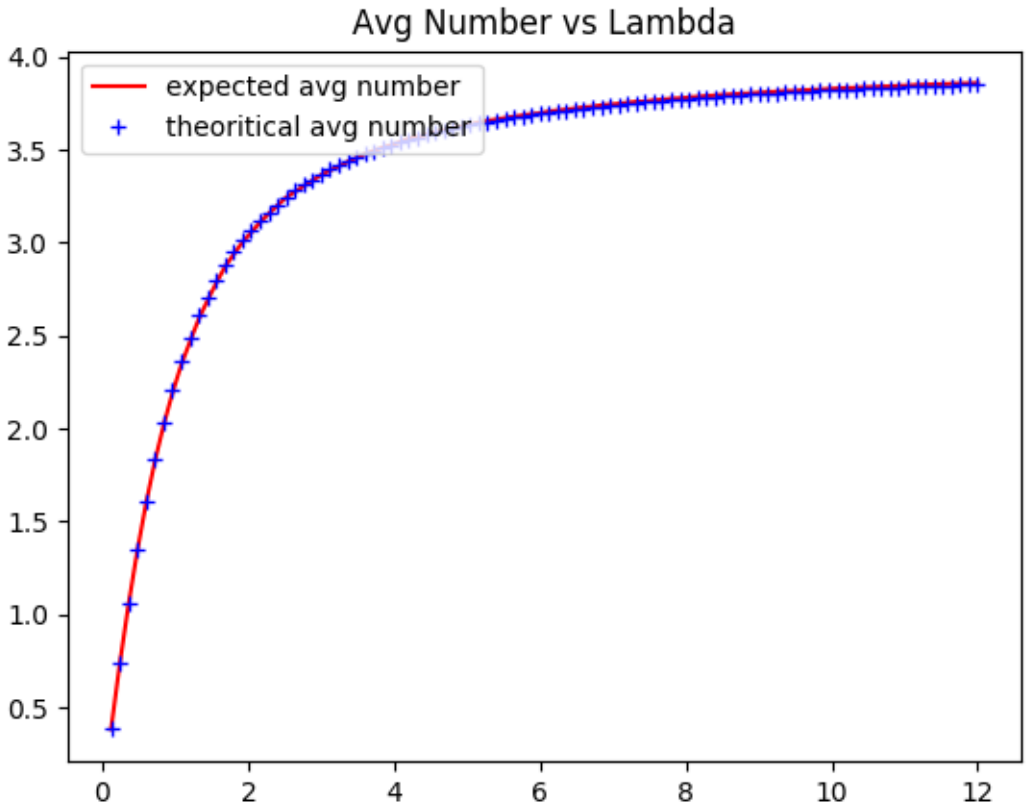


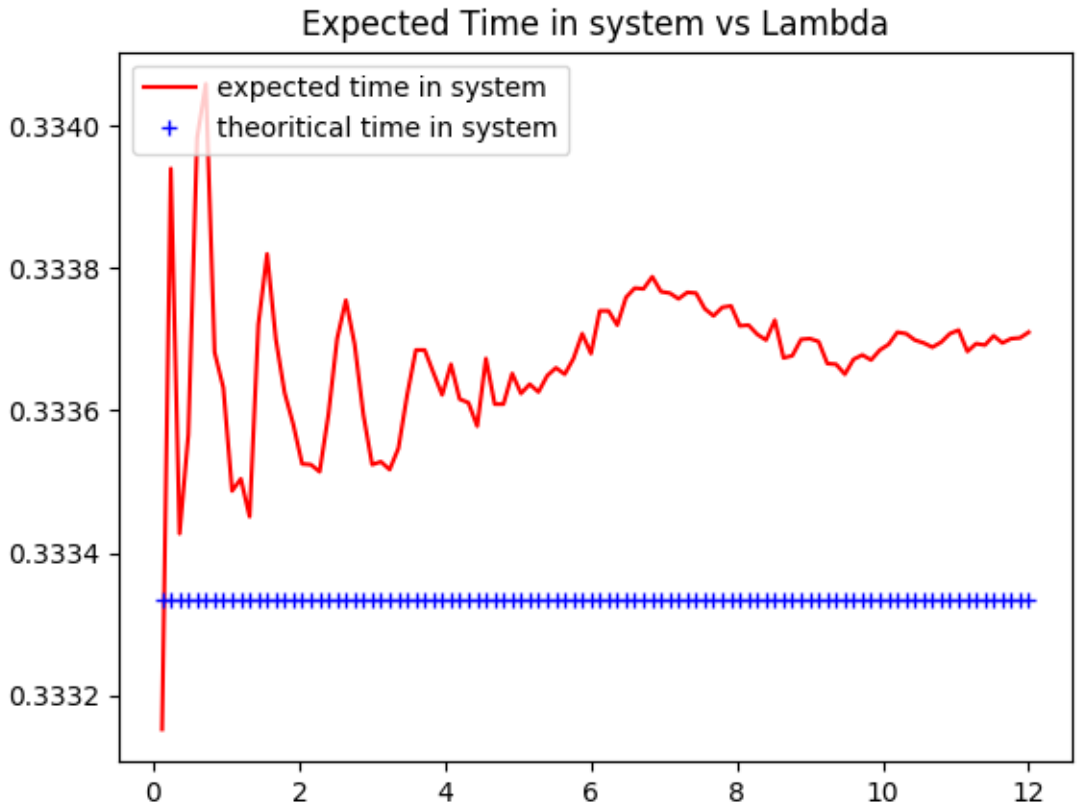

Expected Time in system vs Lambda
expected time in system
theoritical time in system
0.6
0.5
0.4
0.3
0.2
0
1
2
3
4
5
6

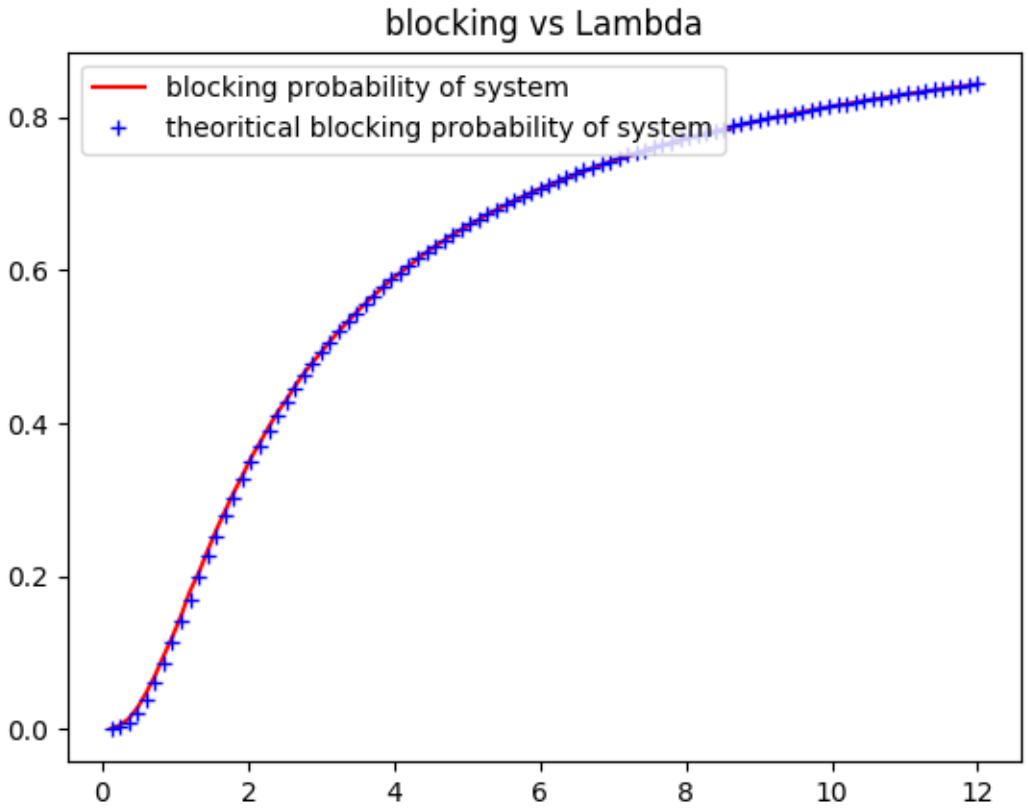

blocking vs Lambda
blocking probability of system
theoritical blocking probability of system
1.00
0.98
0.96
0.94
0.92
0.90
0
1
2
3
4
5
6

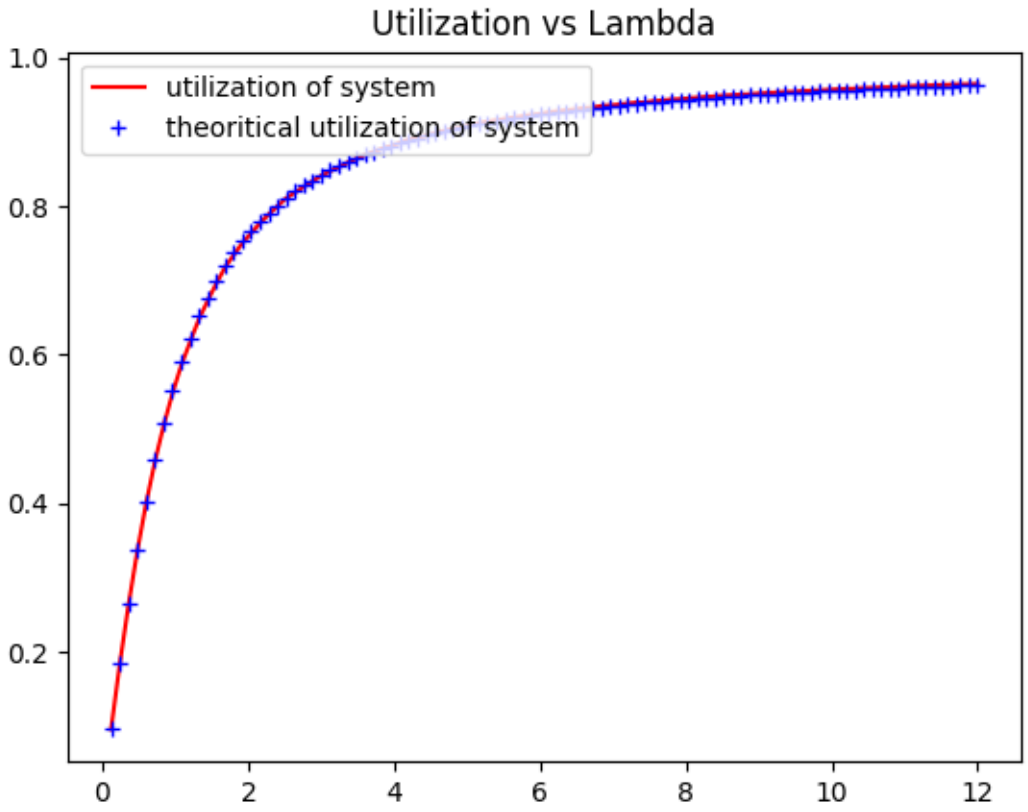

Utilization vs Lambda
utilization of system
theoritical utilization of system
1.00
0.95
0.90
0.85
0.80
0.75
0.70
0
1
2
3
4
5
6

4.4 L10, K4, M4, U3




Expected Time in system vs Lambda
expected time in system
theoritical time in system
0.3340
0.3338
0.3336
0.3334
0.3332
0
2
4
6
8
10
12


blocking vs Lambda
blocking probability of system
theoritical blocking probability of system
0.0
0.2
0.4
0.6
0.8
0
2
4
6
8
10
12


Utilization vs Lambda
utilization of system
theoritical utilization of system
1.0
0.8
0.6
0.4
0.2
0
2
4
6
8
10
12