BU Metropolitan College

BU Metropolitan College

Decision Support System for Healthcare Capacity Planning

Canan Gunes Corlu, PhD, John Maleyeff, PhD, Chenshu Yang, MS, Tianhuai Ma, MS, Yanting Shen, MS Candidate
Administrative Sciences, Metropolitan College, Boston University

Introduction

Capacity management of hospital staff and other resources is an important challenge faced by a healthcare administrator. Because of the variation in service times and the inability to inventory services, capacity buffers are required to ensure reasonable waiting times for patients. Figure 1 shows a generic example that is applicable to any queuing system. As the server utilization increases, waiting time will increase in a pattern commonly known as a hockey stick.

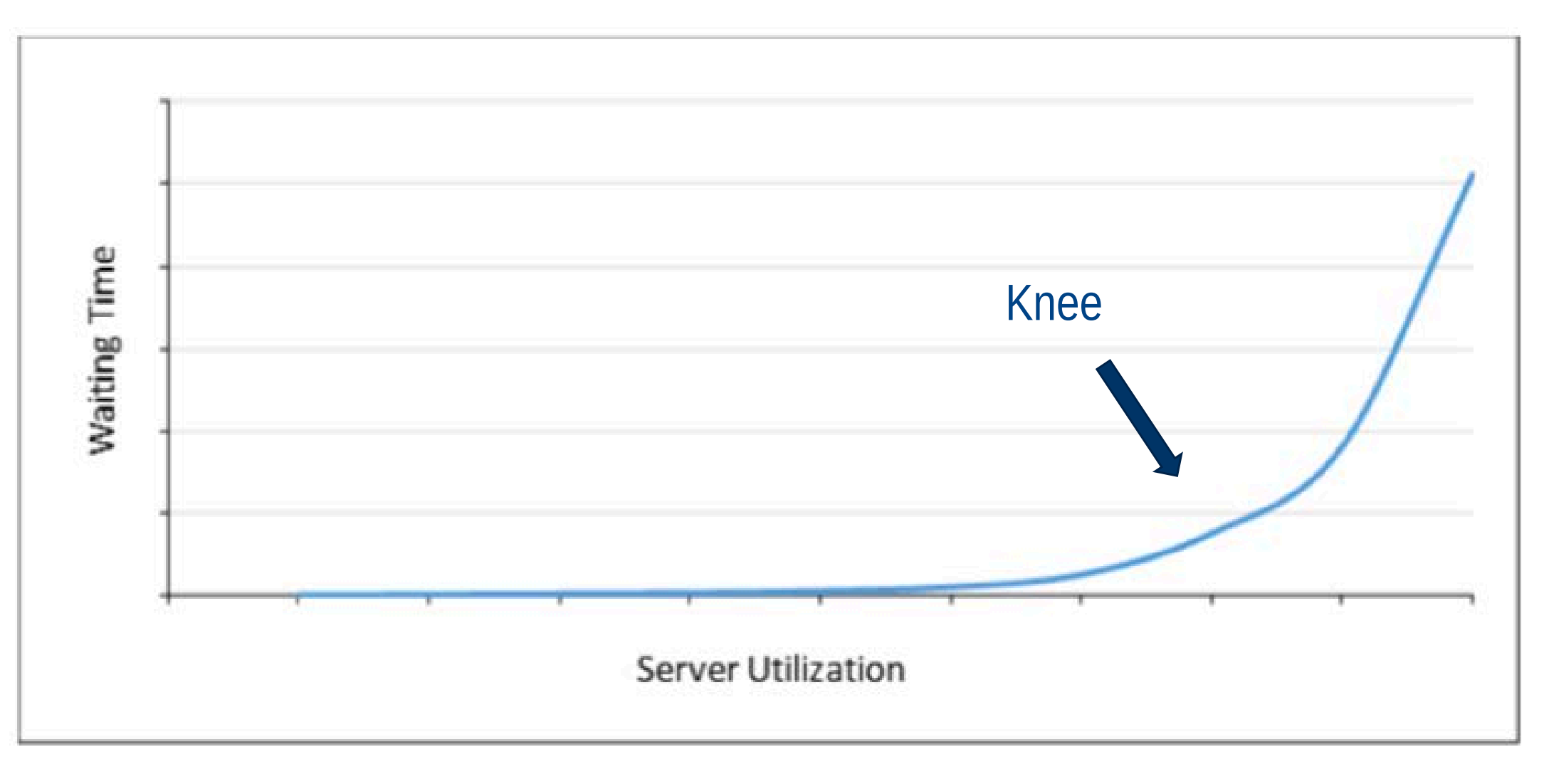


Figure 1 Server Utilization versus Waiting Time “Hockey-Stick.”

A capacity plan will effectively balance the needs of the planner (i.e., by maximizing server utilization) and the needs of the patient (by minimizing waiting times). In some fields, the optimal system configuration takes place at a threshold referred to as the knee of the curve.

This study concerns the development of a decision support system (DSS) using Python to determine optimal capacity buffers using a Monte Carlo simulation (MCS) and a knee optimization model that allows for flexibility in specifying uncertain arrival patterns and service times.

Assumptions

The queuing system is robust with the following structure:

- Single queue served by multiple parallel servers
- Infinite customer population
- Arrivals are deterministic or random
- Infinite queue size
- First-come first-served discipline
- Gamma distributed service time variation

Figure 2 shows gamma distributions for various values of the coefficient variation (CV).

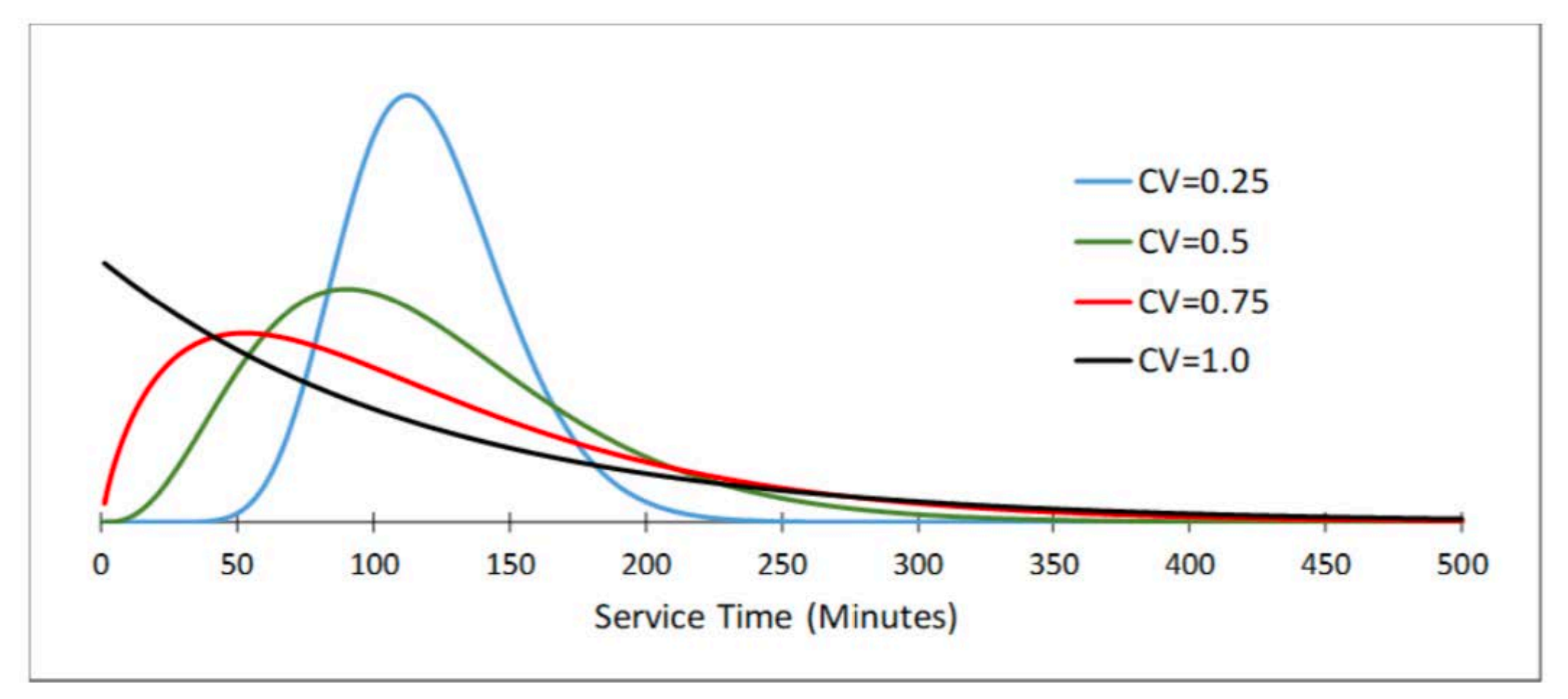


Figure 2 Gamma Distributions (Average Service Time is 120 Minutes).

Methods

Simulation Model:

- The performance statistics were generated based on the server utilization (ρ), which is the ratio of the customer arrival rate (λ) to the system’s service rate.
- The inputs are the number of servers, CV, and the arrival pattern. The outputs are the customer’s wait time in the system (Ws), wait time in the queue (Wq), the number of customers in the system (Ls) and the number of customers in the queue (Lq).

Optimization Model:

- This study uses a Kleinrock’s power function to identify the level of ρ using the equation:

$$Power(\rho) = \frac{\rho}{\lambda\ Ws}$$

- The MCS finds the knee (optimal server utilization) by changing ρ from 40% to 95% (in increments of 5%) and simulating the system repeatedly over this range.

Observations

Congestion and Instability:

- When the system is congested the variation of wait times increases as a percentage of the average wait time, and wait times behaved erratically.
- The variation of the time spent in a congested system exhibited a great deal of instability due to the significant autocorrelation among patient wait times.
- Increasing servers or reducing service times are the approaches to relieve congestion.

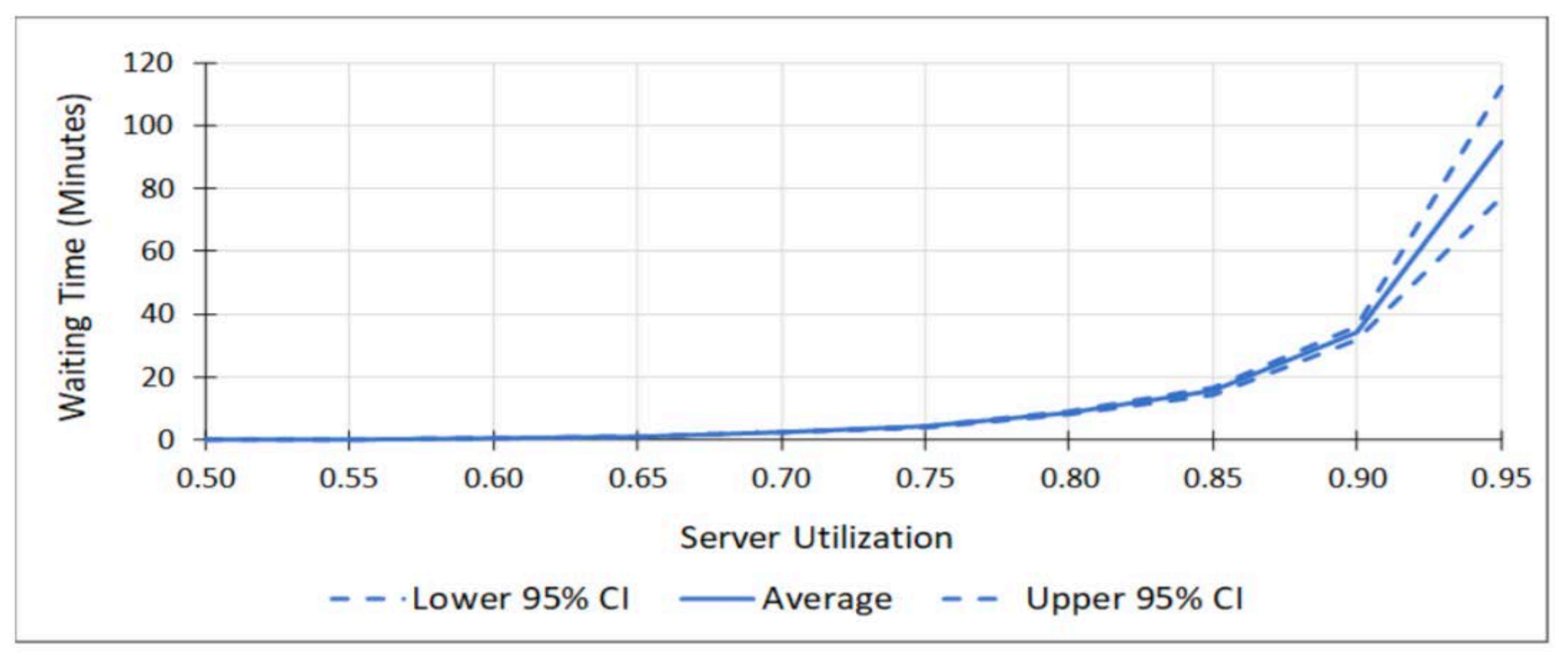
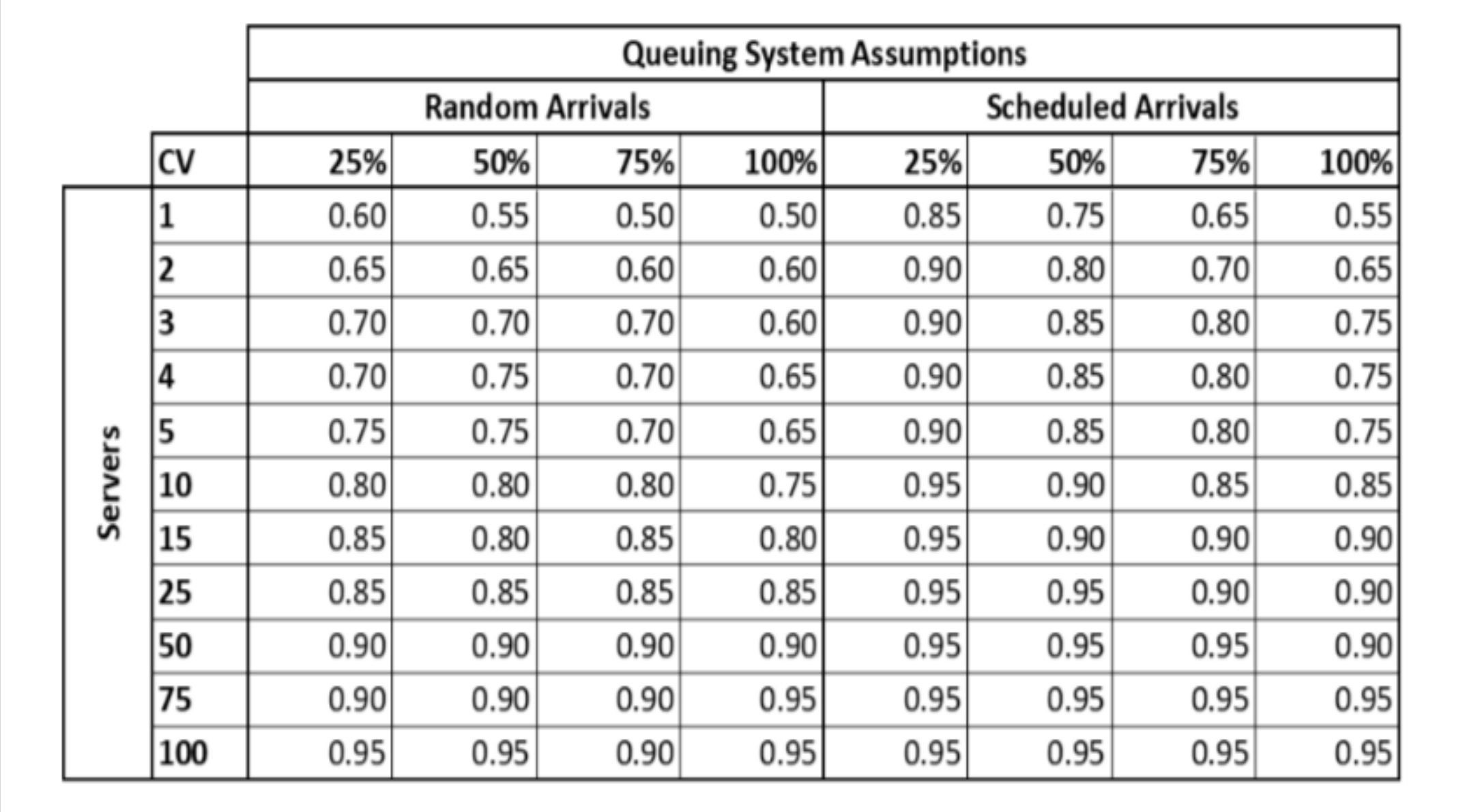


Figure 3 Confidence Intervals (s=15, CV=0.5, 6/hour random arrivals)

Optimal Capacity Buffering:

- A factorial experimental design was used to explore how the knee changed based on levels of key variables. Figure 4 shows the knee for 88 conditions.



		Queuing System Assumptions							
		Random Arrivals				Scheduled Arrivals			
	CV	25%	50%	75%	100%	25%	50%	75%	100%
Servers	1	0.60	0.55	0.50	0.50	0.85	0.75	0.65	0.55
	2	0.65	0.65	0.60	0.60	0.90	0.80	0.70	0.65
	3	0.70	0.70	0.70	0.60	0.90	0.85	0.80	0.75
	4	0.70	0.75	0.70	0.65	0.90	0.85	0.80	0.75
	5	0.75	0.75	0.70	0.65	0.90	0.85	0.80	0.75
	10	0.80	0.80	0.80	0.75	0.95	0.90	0.85	0.85
	15	0.85	0.80	0.85	0.80	0.95	0.90	0.90	0.90
	25	0.85	0.85	0.85	0.85	0.95	0.95	0.90	0.90
	50	0.90	0.90	0.90	0.90	0.95	0.95	0.95	0.90
	75	0.90	0.90	0.90	0.95	0.95	0.95	0.95	0.95
	100	0.95	0.95	0.90	0.95	0.95	0.95	0.95	0.95

Figure 4 Optimal Knee Values

Observations

- When determining the knee for each condition, 10 trials with 10000 iterations each were used to generate results for every level of server utilization.
- Figure 5 shows how the power function is used to determine the knee for 4 of the 88 conditions.

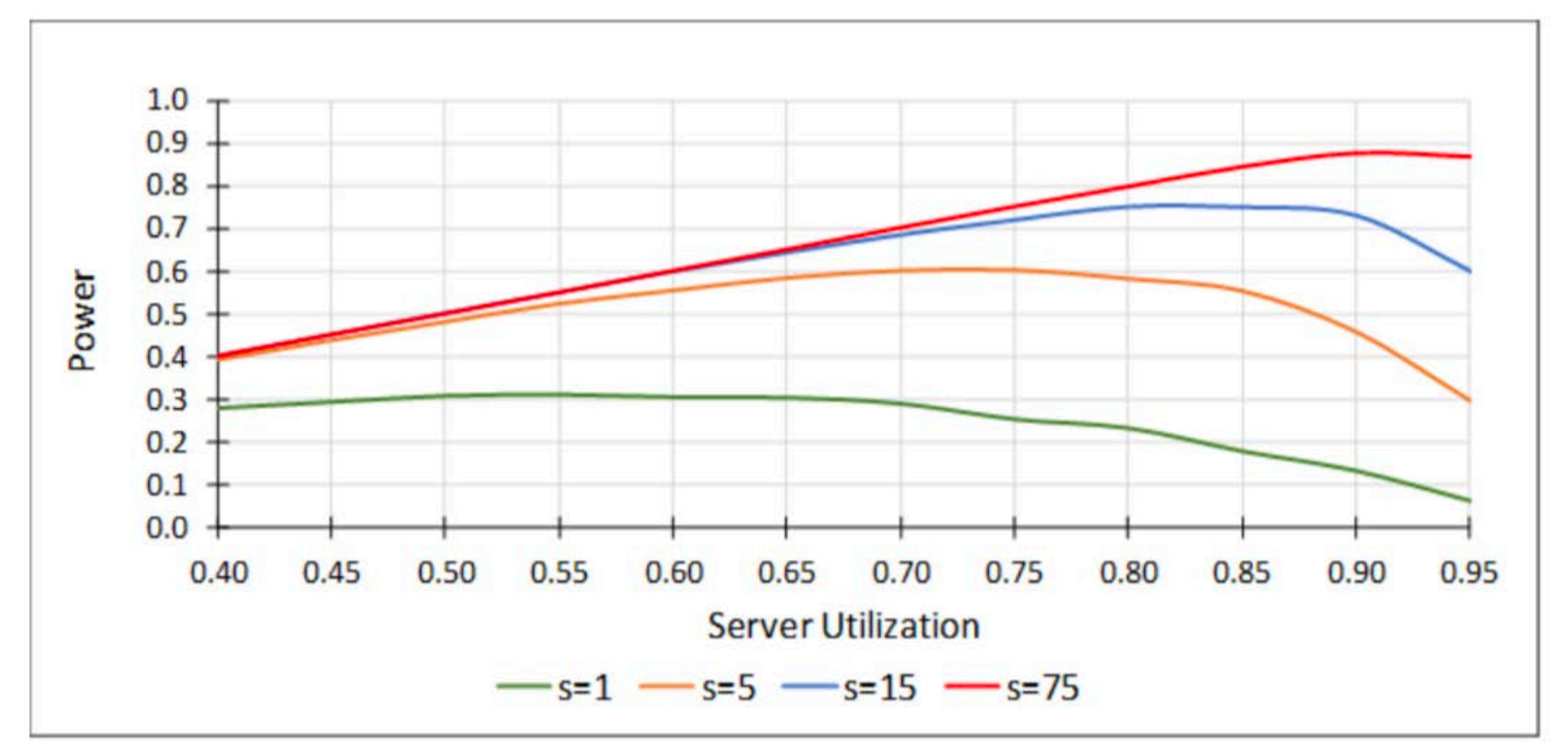


Figure 5 Utilization vs. Power Function (Random arrivals, CV=0.5).

- Figure 6 shows how the service time CV and the pattern of arrivals affected the knee.

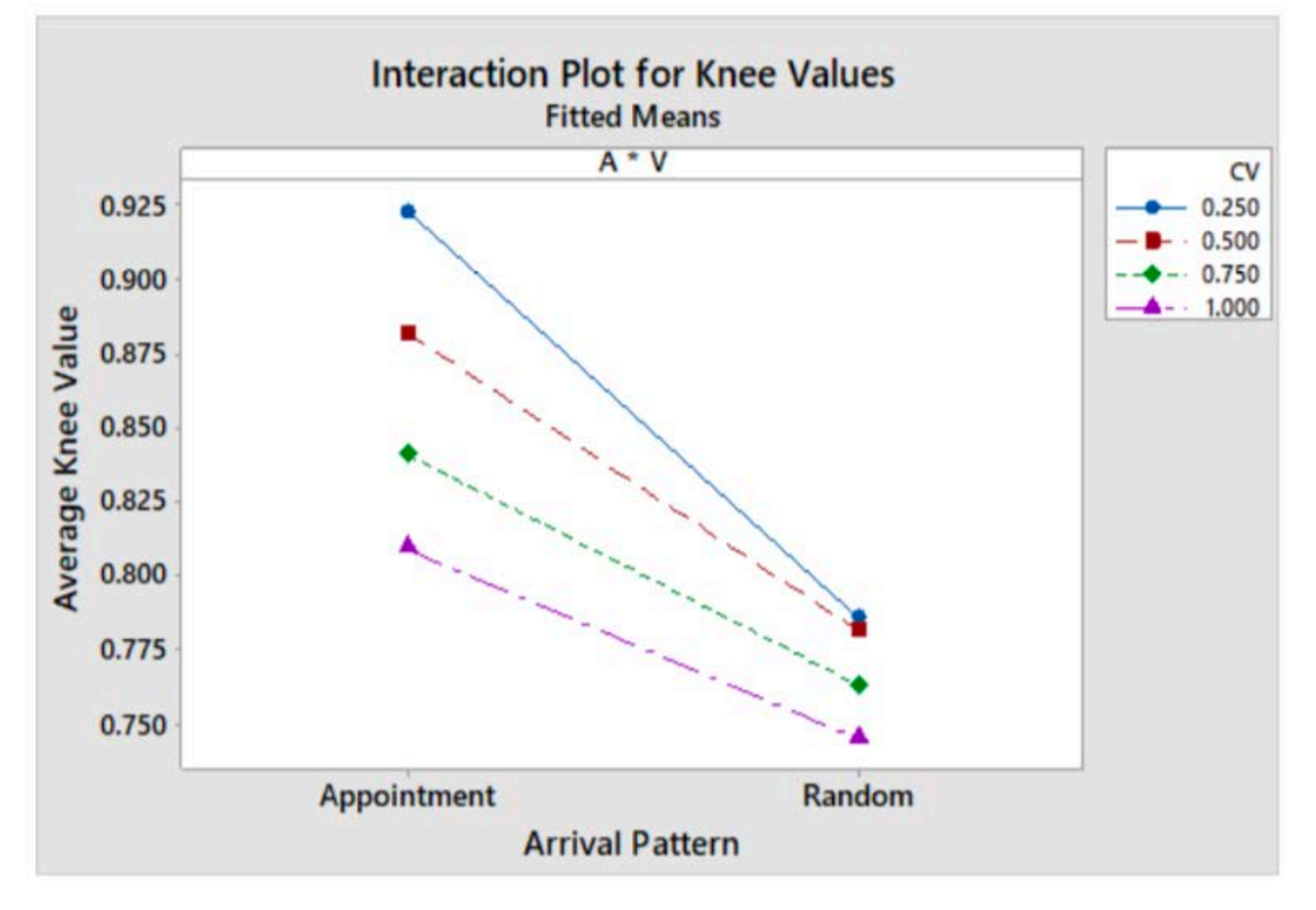
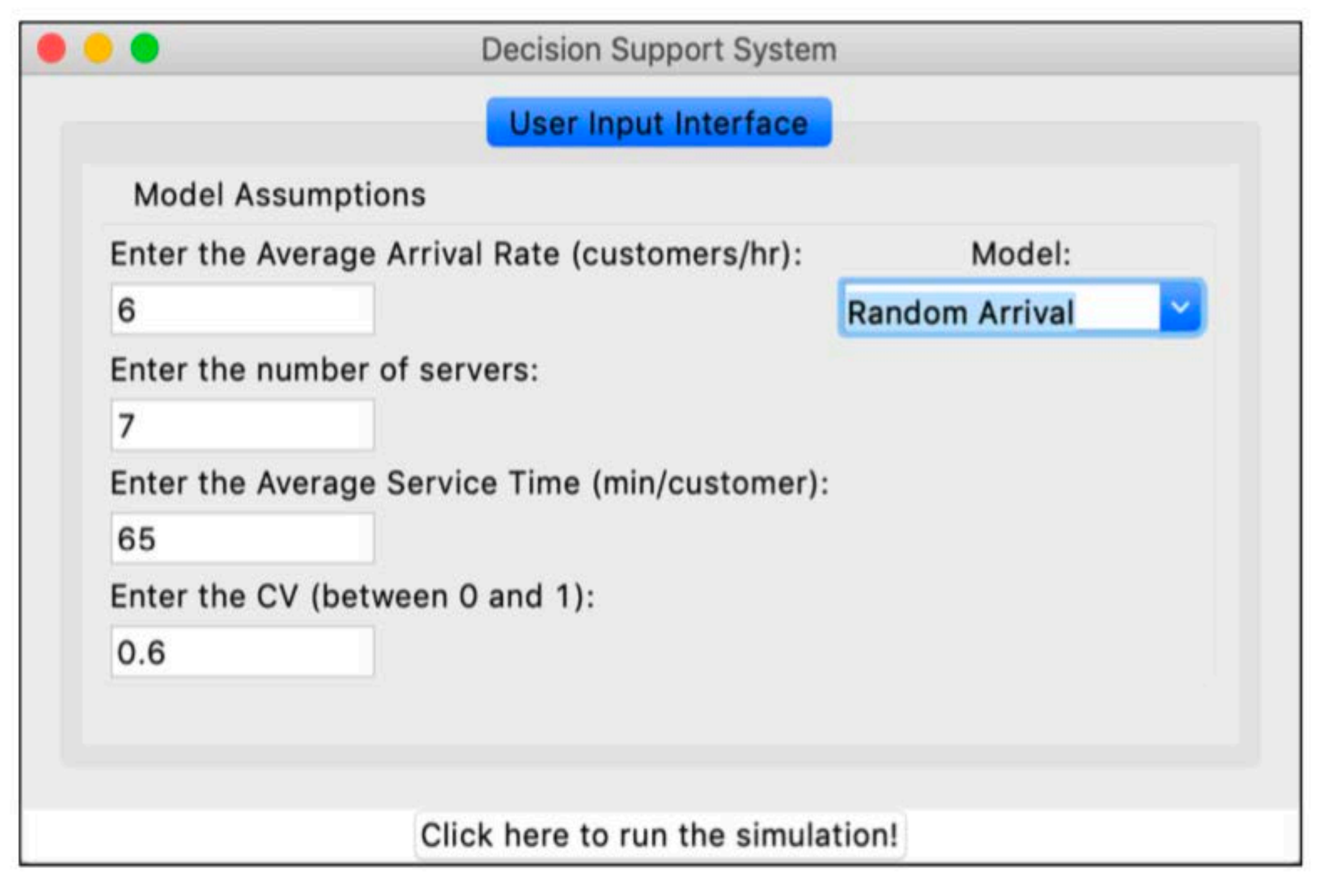


Figure 6 Knee Values versus Arrival Patterns.

Decision Support System - Input

A DSS has been developed in Python. Figure 7 shows the user interface, where the user enters the average service rate, average service time, CV, the number of servers, and a choice of random or scheduled arrivals.



Model Assumptions

Enter the Average Arrival Rate (customers/hr): 6

Model: Random Arrival

Enter the number of servers: 7

Enter the Average Service Time (min/customer): 65

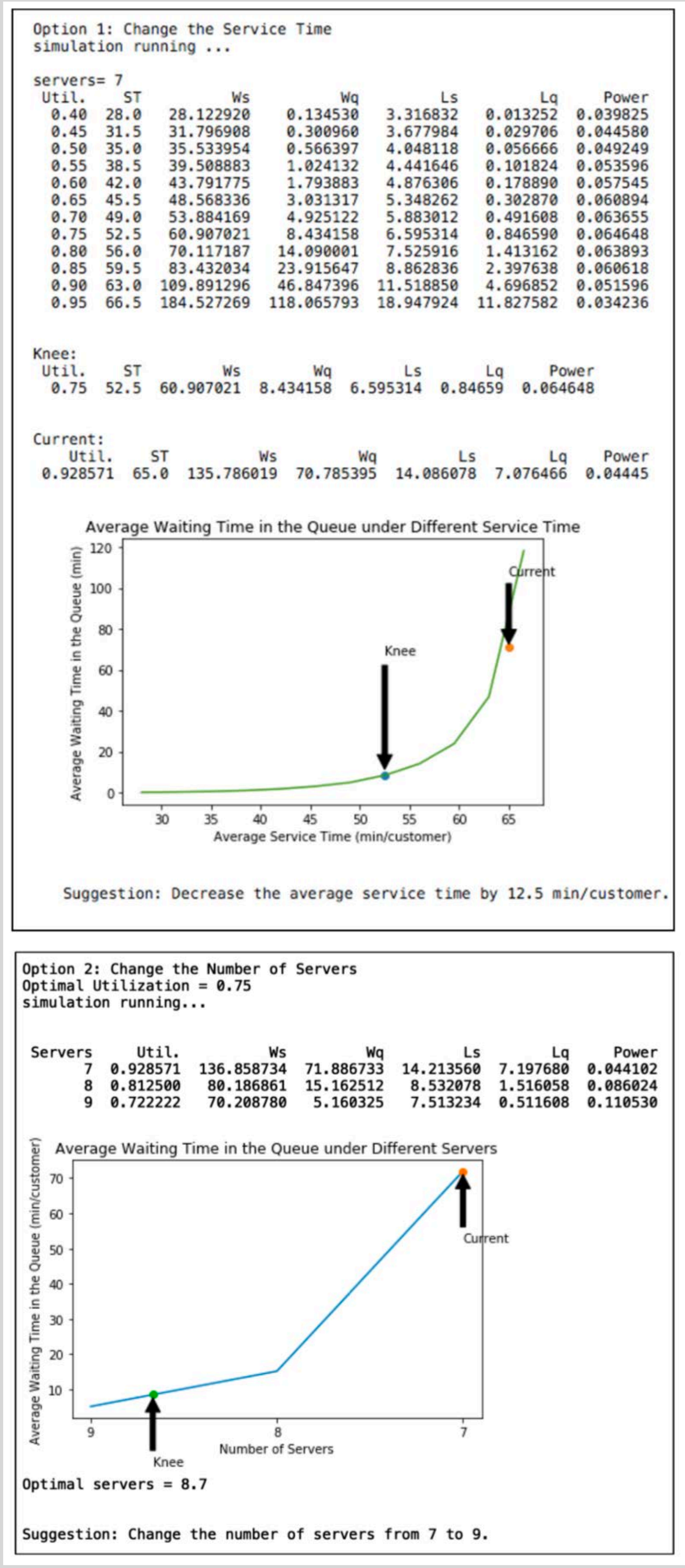
Enter the CV (between 0 and 1): 0.6

Figure 7 DSS User Interface

Decision Support System - Output

The DSS begins by simulating the queuing system based on the user inputs, then it finds the knee using the MCS and knee optimization model.

Figure 8 shows the current server utilization compared to results with a range of utilization values based on service time changes, and an alternative analysis that focuses on changing the number of servers to achieve the optimal system configuration.



```
Option 1: Change the Service Time
simulation running ...

servers= 7
 Util.   ST         Ws          Wq         Ls         Lq     Power
  0.40  28.0   28.122920    0.134530   3.316832   0.013252  0.039825
  0.45  31.5   31.796908    0.300960   3.677984   0.029706  0.044580
  0.50  35.0   35.533954    0.566397   4.048118   0.056666  0.049249
  0.55  38.5   39.508883    1.024132   4.441646   0.101824  0.053596
  0.60  42.0   43.791775    1.793883   4.876306   0.178890  0.057545
  0.65  45.5   48.568336    3.031317   5.348262   0.302870  0.060894
  0.70  49.0   53.884169    4.925122   5.883012   0.491608  0.063655
  0.75  52.5   60.907021    8.434158   6.595314   0.846590  0.064648
  0.80  56.0   70.117187   14.090001   7.525916   1.413162  0.063893
  0.85  59.5   83.432034   23.915647   8.862836   2.397638  0.060618
  0.90  63.0  109.891296   46.847396  11.518850   4.696852  0.051596
  0.95  66.5  184.527269  118.065793  18.947924  11.827582  0.034236


Knee:
 Util.   ST        Ws        Wq        Ls        Lq     Power
  0.75  52.5  60.907021  8.434158  6.595314  0.84659  0.064648


Current:
    Util.   ST          Ws         Wq         Ls        Lq     Power
 0.928571  65.0  135.786019  70.785395  14.086078  7.076466  0.04445
```

Average Waiting Time in the Queue under Different Service Time
Average Waiting Time in the Queue (min)
120
100
80
60
40
20
0
Current
Knee
30
35
40
45
50
55
60
65
Average Service Time (min/customer)

```
 Suggestion: Decrease the average service time by 12.5 min/customer.
```

```
Option 2: Change the Number of Servers
Optimal Utilization = 0.75
simulation running...


 Servers     Util.          Ws         Wq         Ls         Lq     Power
       7  0.928571  136.858734  71.886733  14.213560  7.197680  0.044102
       8  0.812500   80.186861  15.162512   8.532078  1.516058  0.086024
       9  0.722222   70.208780   5.160325   7.513234  0.511608  0.110530
```

Average Waiting Time in the Queue under Different Servers
Average Waiting Time in the Queue (min/customer)
70
60
50
40
30
20
10
Current
9
8
7
Knee
Number of Servers

```
Optimal servers = 8.7


Suggestion: Change the number of servers from 7 to 9.
```

Figure 8 DSS Output Report