Simulation and Modeling (CS302)

Lecture 02: Simulation of Queuing Systems (QS)

Collected and Edited by:
Dr. Mina Younan
Lecturer of Computer Science,
Faculty of Computers and Information, Minia University

Agenda

- Simulation Tables
- Simulation of Queuing Systems
- Single Server Terminologies
- Simulation of Queuing Systems: An Example
- Lab tutorial for QS

Simulation Tables

- We will present several examples of simulations that can be performed by devising a ***simulation table*** either manually or with a spreadsheet.
- These examples provide insight into the methodology of discrete system simulation and the descriptive statistics used for predicting system performance.
- The ***simulation table*** provides a systematic method for tracking system state over time.

Simulation Tables

The simulation by using a simulation table includes three steps:

1. Determine the characteristics of each of the inputs to the simulation. These may be modeled as probability distributions.
2. Construct a simulation table. Each simulation table is different. It depends on the problem at hand.
3. For each repetition i, generate a value for each of the p inputs, and calculate a value of the response y_i. The input values may be computed by sampling values from the distributions determined in step 1. A response depends on the inputs and one or more previous responses.

Here is an example of a simulation table, which works on p inputs, x_{ij}, $j = 1, 2, \dots, p$, and one response, y_i, for each of repetitions $i = 1, 2, \dots, n$.

Simulation Table

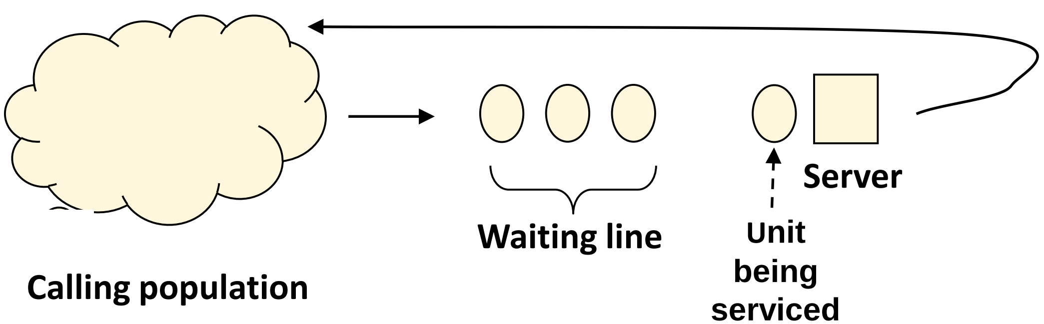
Repetition i	Inputs						Response
	x_{i1}	x_{i2}	…	x_{ij}	…	x_{ip}	y_i
1							
2							
.							
.							
n							

Simulation of Queuing Systems

A queuing system (QS) is described by:

1. Its calling population, the population of potential customers, which may be finite or infinite,
2. The nature of arrivals, which may be scheduled or random,
3. The service mechanism, which may consists of a single server or multiple (parallel/sequential) servers,
4. Its capacity, number of units that can wait in line, which may be limited or unlimited, and
5. The queueing discipline, which may be FIFO, LIFO, random, shortest first, or priority queue.

A simple single-server QS is shown in the figure. The QS includes the **server**, the **unit being serviced** (if one is being serviced), and **units in the queue** (if any are waiting).



Simulation of Queuing Systems

In this single-server queue:

- The calling population is infinite; that is, if a unit leaves the calling population and joins the waiting line or enters service, there is no change in the arrival rate of other units that may need service.
- Arrivals for service occur one at a time in a random fashion.
- Service times are of some random length according to a probability distribution which does not change over time.
- The system capacity has no limit, meaning that any number of units can wait in line.
- Units are served in the order of their arrival, i.e. first-in first-out (FIFO), by a single server.

Simulation of Queuing Systems

In this single-server queue:

- Arrivals and services are defined by the distribution of the *time between arrivals* (*inter-arrival times*) and the distribution of the *service times*, respectively.

- For any simple single- or multi-server queue, the overall effective arrival rate must be less than the total service rate, or the waiting line will grow without bound. When queues grow without bound, they are termed "explosive" or unstable.

Single Server Terminologies

Before introducing several simulations of queuing systems, it is necessary to understand the concept of system state, events, and simulation clock.

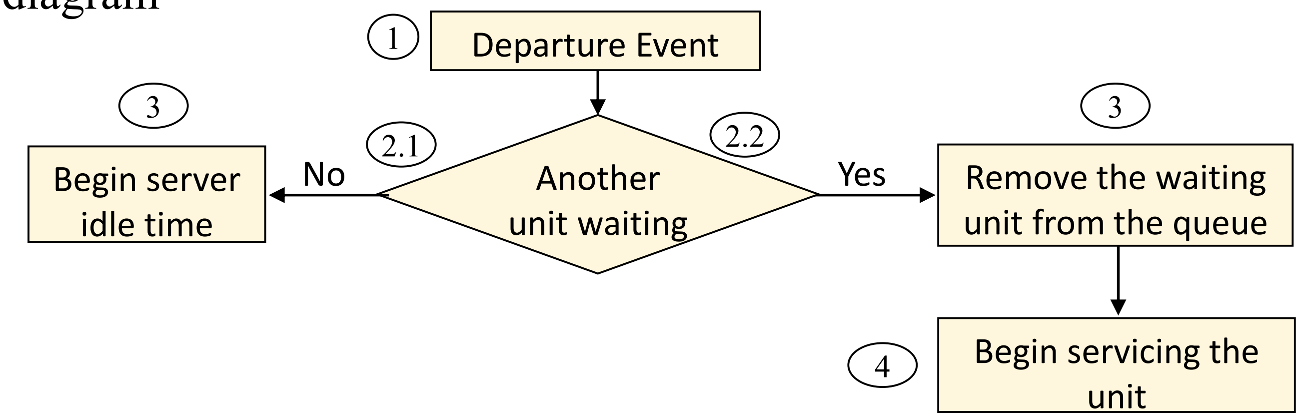
In a queuing system:

- The **state** of the system is the number of units in the system and the status of the server, busy or idle.
- An **event** is a set of circumstances that cause an instantaneous change in the state of the system.
- In a single-server QS there are only two possible events that can affect the state of the system. They are:
  - The ***arrival event***: the entry of a unit into the system.
  - The ***departure event***: the completion of service on a unit.
- The **simulation clock** is used to track simulated time.

Single Server Terminologies

The departure event:

- If a unit has just completed service, the simulation proceeds in the manner shown in the following flow diagram



- Server has only two possible states (***busy*** or ***idle)***. The relationship of these two outcomes to the status of the queue is shown in the following table

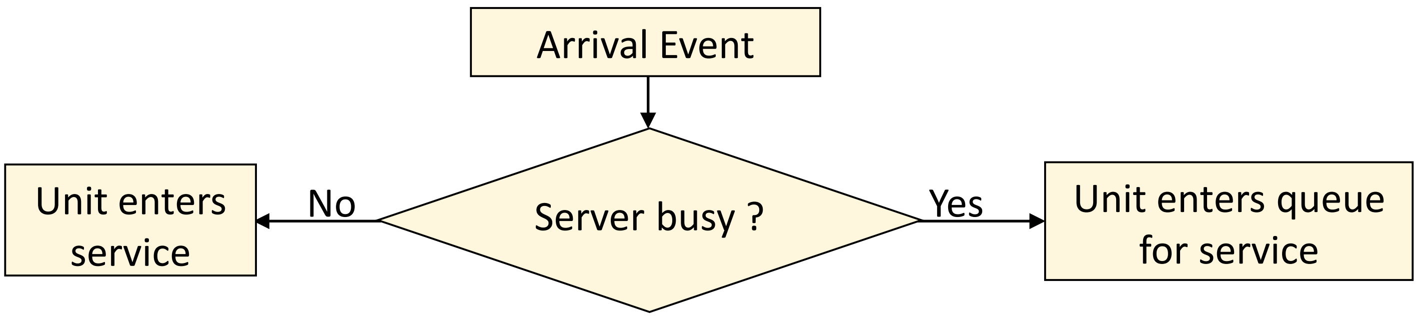
- If the queue is not empty, another unit will enter the server and it will be **busy**. If the queue is empty, the server will be **idle** after a service is completed.

	Queue Status	
	Not Empty	Empty
Server outcome	Busy	Idle

Single Server Terminologies

The arrival event:

- The arrival event occurs when a unit enters the system. The flow diagram for the arrival event is shown in figure below



- The unit may find the server either idle or busy; therefore, either the unit begins service immediately, or it enters the queue for the server. Upon arrival the unit follows the course of actions shown in the table below

- If the server is busy, the unit enters the queue. If the server is idle and the queue is empty, the unit begins service. It is not possible for the server to be idle and the queue to be nonempty.

		Queue Status	
		Not Empty	Empty
Server status	Busy	Enter queue	Enter queue
	Idle	**Impossible**	Enter service

Single Server Terminologies

- Simulations of QSs generally require the maintenance of an ***event list*** for determining what happens next. The event list tracks the future times at which the different types of events occur.
- Simulation clock times for arrivals and departures are computed in a simulation table customized for each problem.
- In simulation, events usually occur at random times (uncertainty in real life). E.g., it is not known when the next customer will arrive at a grocery checkout counter, or how long the bank teller will take to complete a transaction.
- In these cases, a statistical model of the data is developed from either data collected and analyzed, or subjective estimates and assumptions.
- Random numbers are distributed uniformly and independently on the interval (0,1).
- In a single-server QS, inter-arrival times and service times are generated from the distributions of these random variables. Examples discussed next show how such times are generated.

Single Server Terminologies

Examples:

- Consider a service facility with a single server. Assume 6 customers arrived at the facility one at a time.
- For simplicity, assume that the times between arrivals are in the range 1 to 6 and were generated by rolling a die five times and recording the up face.→ (random (1:6))
- Table in the next slide contains a set of five inter-arrival times generated in this manner. These five inter-arrival times are used to compute the arrival times of six customers at the QS.
- Note that, in general, for n customers we generate n-1 inter-arrival times.

Simulation of Queuing Systems: An Example

<u>Customers' arrival details:</u> The 1st customer is assumed to arrive at clock time 0. This starts the clock in operation. After two time units (seconds, minutes … etc.), the 2nd customer (i.e., at a clock time of 2). The 3rd customer arrives four time units later (i.e., at a clock time of 6); and so on. → Random (1:6)

Inter-arrival times and clock times

Customer	Interarrival Time	Arrival Times on Clock
1	-	0
2	2	2
3	4	6
4	1	7
5	2	9
6	6	15

The second time of interest is the service time. This table contains service times generated at random from a distribution of service times. The only possible service times are 1, 2, 3, and 4 time units. → Random (1:4)

Service Times

Customer	1	2	3	4	5	6
Service Time	2	1	3	2	1	4

Simulation of Queuing Systems: An Example

- Assuming that all four values are equally likely to occur, these values could have been generated by placing the numbers 1 through 4 on chips and drawing the chips from a hat with replacement, being sure to record the numbers selected.
- Now, the inter-arrival times and service times must be meshed to simulate the single-server QS.

Simulation table for the given problem

	A	B	C	D	E
1	Customer Number	Arrival Time (Clock)	Time Service Begins (Clock)	Service Time (Duration)	Time Service Ends (Clock)
2	1	0	0	2	2
3	2	2	2	1	3
4	3	6	6	3	9
5	4	7	9	2	11
6	5	9	11	1	12
7	6	15	15	4	19

Simulation of Queuing Systems: An Example

- As shown in the table of “Inter-arrival times”, the first customer arrives at clock time 0 and immediately begins service, which requires two minutes. Service is completed at clock time 2. The second customer arrives at clock time 2 and is finished at clock time 3. → no wait/delay , server do services in the time (busy)
- Note that the fourth customer arrived at clock time 7, but service could not begin until clock time 9. This occurred because customer 3 did not finish service until clock time 9.
- Simulation table in the previous slide is designed specifically for a single-server QS, which serves customers on FIFO basis. It keeps track of the clock time at which each event occurs. The second column records the clock time of each arrival event, while the last column records the clock time of each departure event.

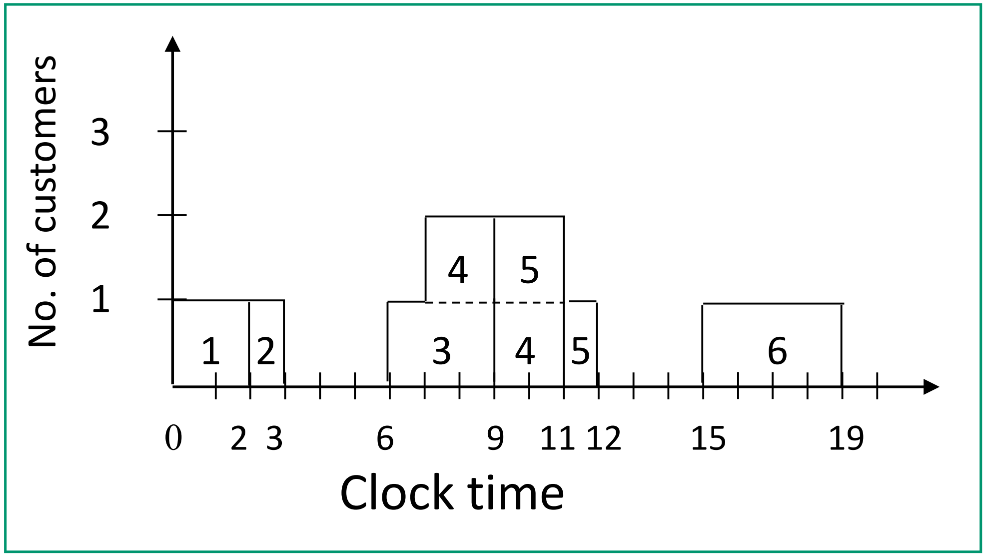
Simulation of Queuing Systems: An Example

- The occurrence of the two types of events in chronological order (i.e., ordered by clock time) is shown in the following table and figure.

Chronological Ordering of Events

Event Type	Customer Number	Clock Time
Arrival	1	0
Departure	1	2
Arrival	2	2
Departure	2	3
Arrival	3	6
Arrival	4	7
Departure	3	9
Arrival	5	9
Departure	4	11
Departure	5	12
Arrival	6	15
Departure	6	19

customers in the system → wait/delay level



This chart depicts the number of customers in the system at the various clock times. It is a visual image of the event listing of Table

Simulation of Queuing Systems: An Example

- Customer 1 in the system takes 2 minutes (from clock time 0 to clock time 2). Customer 2 arrives at clock time 2 and departs at clock time 3 (i.e.,).
- No customers are in the system from clock time 3 to clock time 6. → server is in ideal state
- During some time periods two customers are in the system, such as at clock time 8, when both customers 3 and 4 are in the system.
- Also, there are times when events occur simultaneously, such as at clock time 9, when customer 5 arrives and customer 3 departs.

Lab Tutorial for QS

Queuing System

Customer Number	Interarrival Time	Arrival Time (Clock)	Service Code	Service Title	Time Service Begins (Clock)	Service Time (Duration)	Time Service Ends (Clock)	System State	Customer State
1	0	0	6	Inquiry	0	3	3	Busy	InService
2	7	7	4	Withdrow	7	7	14	4	InService
3	2	9	1	Open	14	10	24	Busy	5
4	2	11	4	Withdrow	24	7	31	Busy	13
5	5	16	6	Inquiry	31	3	34	Busy	15
6	9	25	2	Delete	34	15	49	Busy	9

Ser.ID	Service	Ser.Duration
1	Open	10
2	Delete	15
3	Deposite	5
4	Withdrow	7
5	Transfer	8
6	Inquiry	3
7	Report	4

<u>Simulation Table for:</u>

1. Modeling QS behavior over the time
2. Starting or initiating constants/attributes

Lab Tutorial for QS

	A	B	C	D	E	F	G	H	I	J	K
1											
2							Queuing System				
3											
4		Customer	Interarrival	Arrival Time	Service		Time Service	Service Time	Time Service	System	Customer
5		Number	Time	(Clock)	Code	Title	Begins (Clock)	(Duration)	Ends (Clock)	State	State
6		1	0	0	6	Inquiry	0	3	3	Busy	InService
7		2	7	7	4	Withdrow	7	7	14	4	InService
8		3	2	9	1	Open	14	10	24	Busy	5
9		4	2	11	4	Withdrow	24	7	31	Busy	13
10		5	5	16	6	Inquiry	31	3	34	Busy	15
11		6	9	25	2	Delete	34	15	49	Busy	9

<u>Equations of Main simulation table:</u>

- Customer: start cell B6=1, B7=B6+1, and generalize this formula for the all the remaining cells in column B.
- Interarrival (inter-arrival): first customer → C6=0, C7=int(rand()*10) and generalize formula of C7 for all the remaining cells in column C.
    - Int → get integer value, rand() → get random value between (0,1), rand()* 10 → to get value in the recommended range [0,10[. In case or [0,10] → rand()*11

Lab Tutorial for QS

Queuing System

Customer Number	Interarrival Time	Arrival Time (Clock)	Service Code	Service Title	Time Service Begins (Clock)	Service Time (Duration)	Time Service Ends (Clock)	System State	Customer State
1	0	0	6	Inquiry	0	3	3	Busy	InService
2	7	7	4	Withdrow	7	7	14	4	InService
3	2	9	1	Open	14	10	24	Busy	5
4	2	11	4	Withdrow	24	7	31	Busy	13
5	5	16	6	Inquiry	31	3	34	Busy	15
6	9	25	2	Delete	34	15	49	Busy	9

<u>Equations of Main simulation table:</u>

- Arrival time (Clock): D6=0, D7=D6+C7, and generalize formula of D7 for all the remaining cells in column D.
- Service Code: get random value to represent code of the customer service, in this example number of services =7 → E6: (rand ()*7)+1 → "+1" to ensure that value 0 is replaced by 1. because rand ()*7 → represent values under 7 but in this case "+1" may produce code =7.

Lab Tutorial for QS

	A	B	C	D	E	F	G	H	I	J	K
1											
2							Queuing System				
3											
4		Customer	Interarrival	Arrival Time	Service		Time Service	Service Time	Time Service	System	Customer
5		Number	Time	(Clock)	Code	Title	Begins (Clock)	(Duration)	Ends (Clock)	State	State
6		1	0	0	6	Inquiry	0	3	3	Busy	InService
7		2	7	7	4	Withdrow	7	7	14	4	InService
8		3	2	9	1	Open	14	10	24	Busy	5
9		4	2	11	4	Withdrow	24	7	31	Busy	13
10		5	5	16	6	Inquiry	31	3	34	Busy	15
11		6	9	25	2	Delete	34	15	49	Busy	9

Equations of Main simulation table:

- Service title: search for title with service code and write matched service: =LOOKUP(E6,$M$5:$M$11,$N$5:$N$11) → Main parts of this formula,
  - E6 is the lookup code to search for,
  - $M$5:$M$11: range to search in it for the code
  - $N$5:$N$11: range to return correponding value to the code
  - $column symbol$ to make lookup range static in case for formula generalization.

Lab Tutorial for QS

Queuing System

Customer Number	Interarrival Time	Arrival Time (Clock)	Service Code	Service Title	Time Service Begins (Clock)	Service Time (Duration)	Time Service Ends (Clock)	System State	Customer State
1	0	0	6	Inquiry	0	3	3	Busy	InService
2	7	7	4	Withdrow	7	7	14	4	InService
3	2	9	1	Open	14	10	24	Busy	5
4	2	11	4	Withdrow	24	7	31	Busy	13
5	5	16	6	Inquiry	31	3	34	Busy	15
6	9	25	2	Delete	34	15	49	Busy	9

<u>Equations of Main simulation table:</u>

- Service time: search for title with service code and write matched service duration: = LOOKUP(E6,$M$5:$M$11,$O$5:$O$11)
- Time Service (Begins): G6=0, G7= =MAX(I6,D7)
- Time Service (Ends): =SUM(G6:H6)
- System (state): J6="Busy", J7==IF(G7>I6,G7-I6,"Busy")
- Customer (state): K6==IF(D6<G6,G6-D6,"InService")

Lab Tutorial for QS

<u>Equations of Main simulation table:</u>

- Extra for system behavior analysis
- Suppose that C11= number represent minutes and G2 = start time e.g., “08:30 AM” So in case of calculating the clock time, we need to add interval time in minutes to the start time for the first customer, the main idea is to convert integer value that represents minutes into time then add cell such that =TIME(0,C11,0)+G2

# waiting Customers=K15	=COUNTIF(Table1[Customer],"<>InService")
Prop. Of Waiting	K15/COUNT(Table1[Customer])
Total Waiting	…
Avg. Waiting=	…