COMP9334
Capacity Planning for Computer Systems and Networks

Week 7: Discrete event simulation (2)

Last week

- Two topics
  - How to perform discrete event simulation of queues
  - How to general random numbers of a given probability distribution by using inverse transform method

- You should be able to simulate
  - Many types of queues
    - Single-server or multi-server
    - Different queueing disciplines
  - Many inter-arrival time and service time distributions

- However, there are a number of problems …

Problem: data interpretation, simulation length

- Figure from
- Week 6's revision problem #1

Horizontal line - theoretical value; circles - simulated values
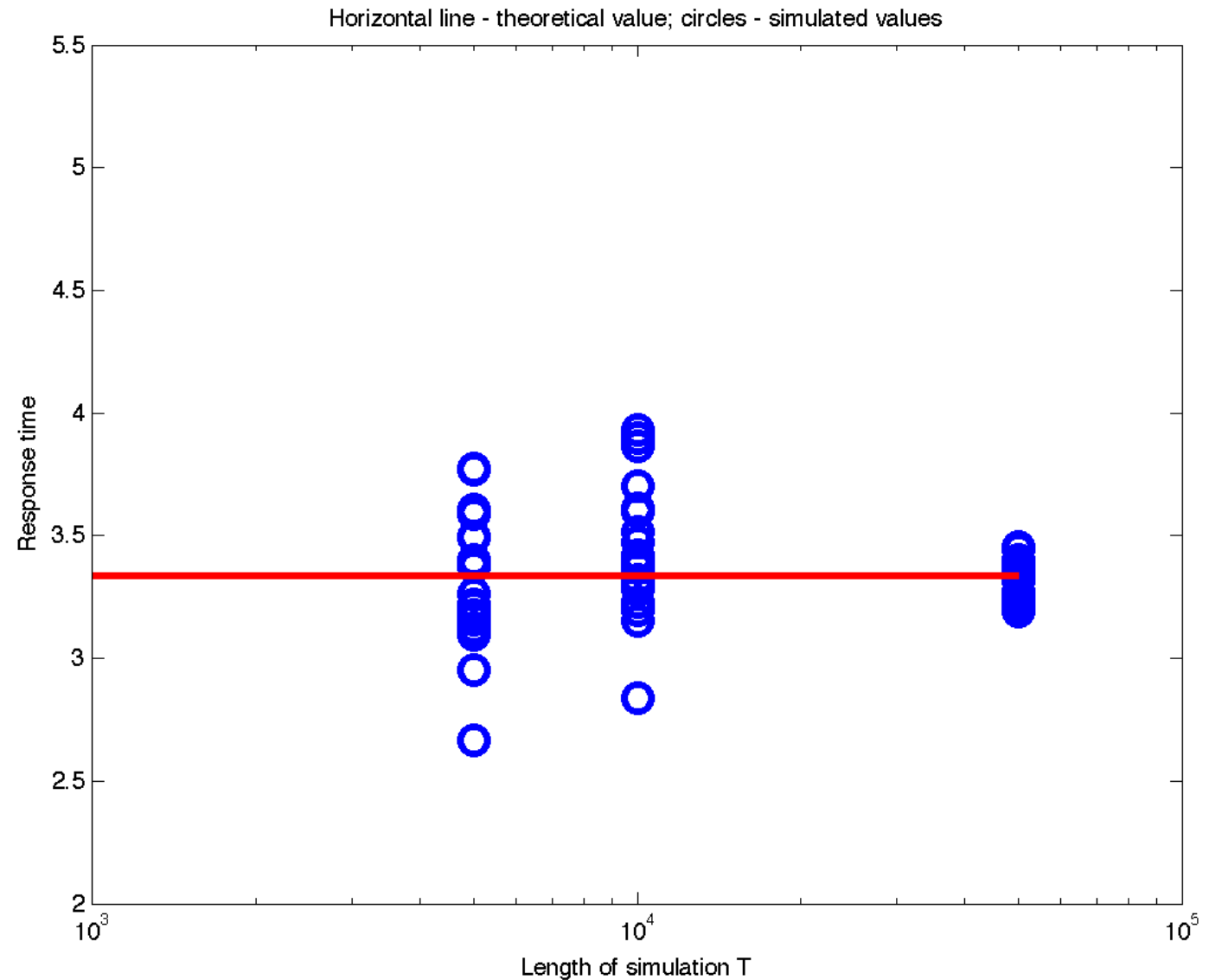
Problem: How do we compare 2 alternative choices?

- Week 6's Revision Problem, #2
- From Queueing theory, we expect the M/M/1 system to be better but simulation doesn't seem to suggest that?
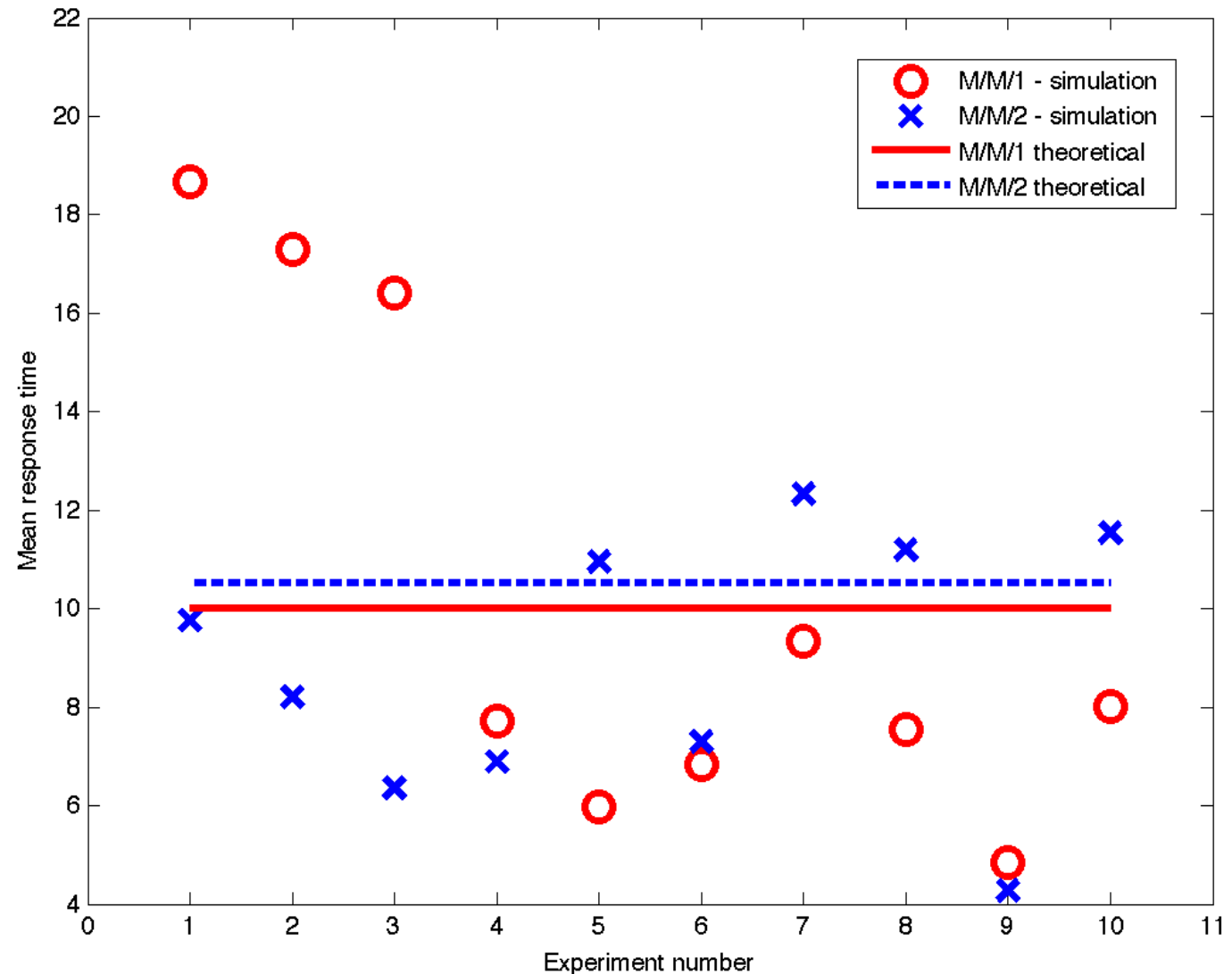
Analysis of simulation results

- A very important topic but it is very often ignored

- Simulation is *not* simply about
  - Writing a computer program
  - Verifying the correctness of the program
  - Running the simulation once and present the results

- Verifying the correctness of the simulation program is important

- It is equally important to do sound statistical analysis on the simulation results obtained

This lecture

- Analysis of simulation results
- How to choose simulation parameters?
    - How long should I simulate for?
    - How many times should I repeat the simulation?
- How do I compare two alternative systems?
- Variance reduction technique
- Simulation tools

Analysis of simulation data

- There are many statistical methods to analyse data depending on the situation

- We will focus on analysing *steady state mean value* only

- For example, we are interested to find the *steady state mean response time* of a queue

- Recall that we talked about
  - Transient and steady state behaviour of queue in Week 3
  - Steady state of Markov chain in Week 4
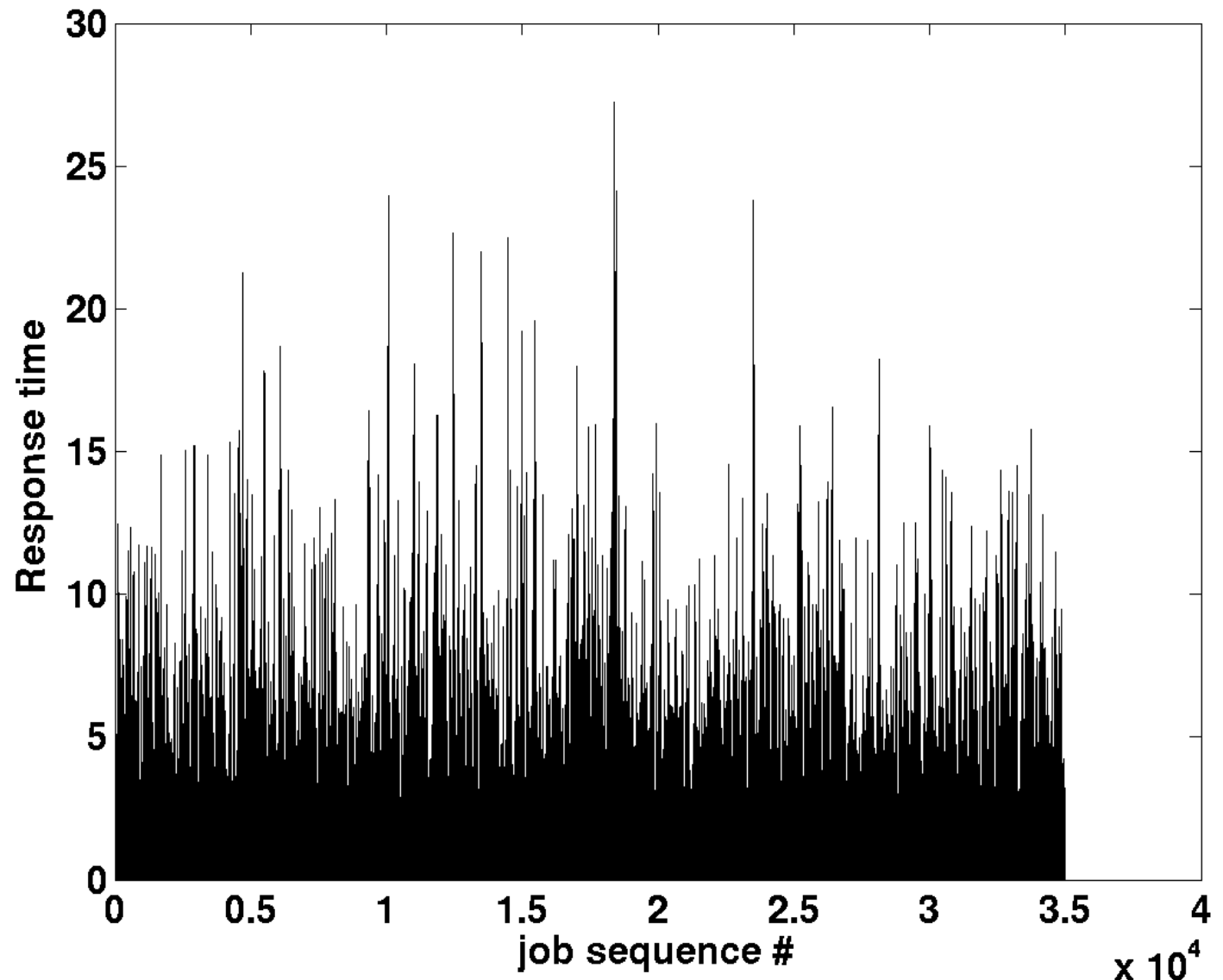
What is steady state? (1)

- Let us simulate an M/M/1 queue with
  - Arrival rate $\lambda = 0.7$
  - Service rate $\mu = 1$
  - Simulation ends when master clock is 50000s
- In this simulation, we record the response time for each job
  - Let $X(k)$ = Response time of kth job
  - The next page shows $X(k)$ changes continuously
- Let N denote the number of jobs in the simulation
  - N = 35000 for our simulation
- Last week, we computed the mean response time using

$$\frac{X(1)+X(2)+...+X(N)}{N}$$

Response time continuously changes over time

- This graph shows response time of X(k) of the *k*-th job where k = 1 to 35000

- Note response time continuously varies

- Response time *does not* settle to a constant value

- But *mean* response time *does* settle

What is steady state? (2)

- Let us instead compute the running mean M(k) where

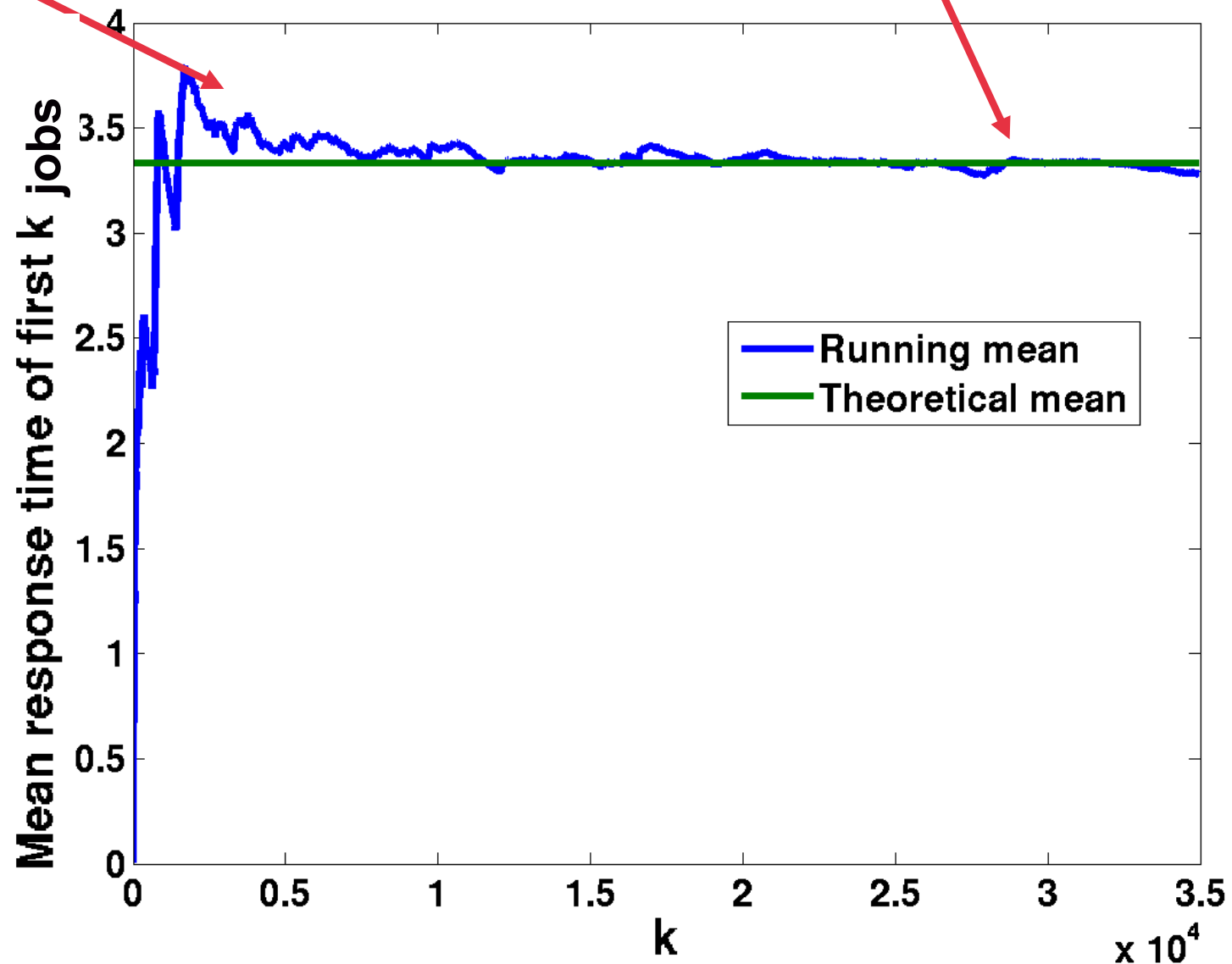$$M(k) = \frac{X(1)+X(2)+...+X(k)}{k}$$

- For example, if k = 5, then

$$M(5) = \frac{X(1)+X(2)+...+X(5)}{5}$$

- Thus M(5) is the mean response time of the first 5 jobs
- In general, M(k) is the mean response time of the first *k* jobs
- Let us plot M(k) - see the next slide

Transient behaviour versus steady state behaviour

Transient removal: Introduction (1)

- The early part of the simulation displays transient (= non-steady state behaviour)

- The later part of the simulation converges or fluctuates around the steady state value

- Since we are interested in the steady state value, we should not use the transient part of the data to compute the steady state value

- We should remove the transient part and only use the steady state part to compute the mean

- One method to identify the transient part is to use visual inspection

  - Note: In the previous slide, we have the theoretical value to guide us but in practice you don't, you will learn a transient removal method based on batch means in your Revision Problem this week

Transient removal: Introduction (2)

- Let us assume that the first *m* jobs constitute the transient part and there are *N* jobs altogether, we should revise the formula to compute the mean to

$$\frac{X(m+1) + X(m+2) + \ldots + X(N)}{N - m}$$

- Note: We used too simple a method to compute the mean last week but I didn't want to complicate things at that time!

- **Important: You must run the simulation long enough so that you have a good number of data points (or jobs) in the steady state part.**

- In order to remove transient properly, you need to understand **replications**

Independent replications

- Assume that we carry out simulations to find out what the steady state mean response time of a queueing system is

  - Important note: *We cannot get exact answer from simulation*

  - We express our simulation results as e.g. there is a probability of 95% that the mean response time is in the interval [3.1,3.3].
  - We call the interval [3.1,3.3] the 95% *confidence interval*.

- Independent replications: Repeat the simulation a number of times using ***different*** sets of random numbers

- Why independent replications?
  - Independent replications allow us to use statistical method to estimate a confidence interval of steady state mean response time
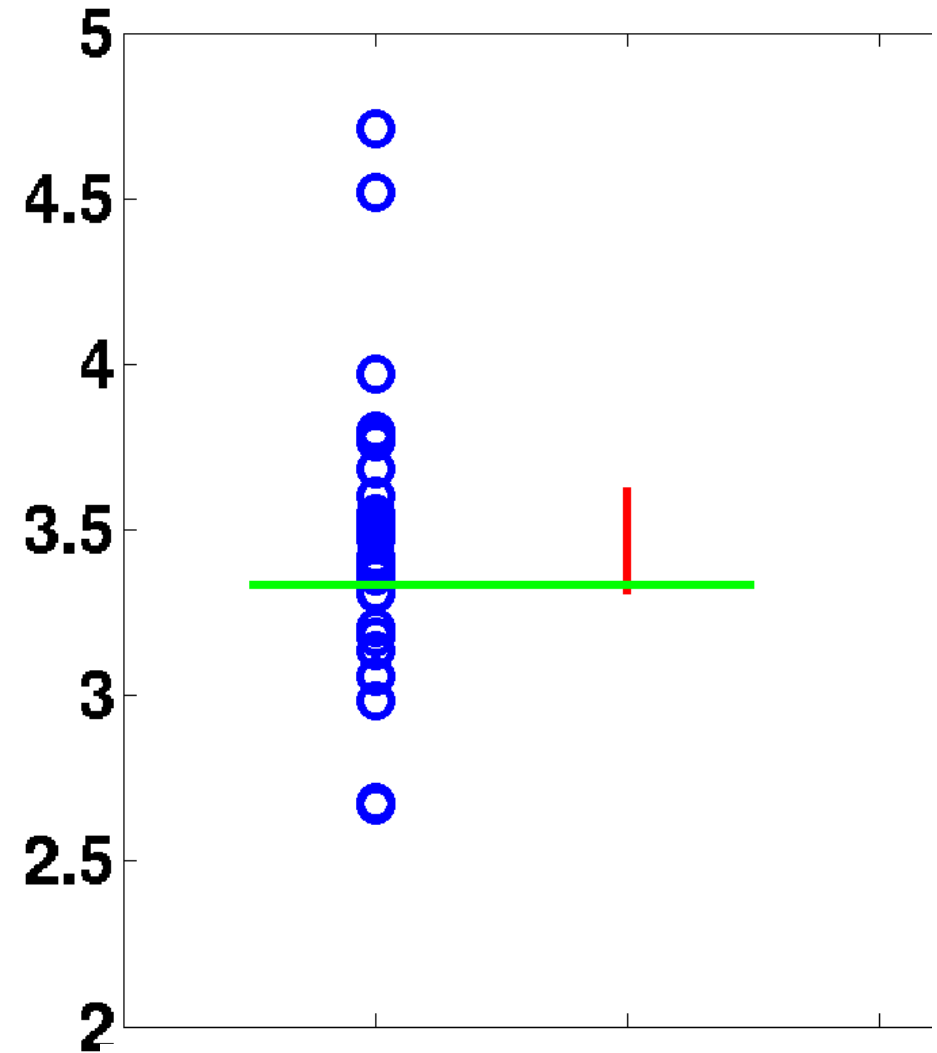
Example: Independent replications

- We want to use simulation to estimate the mean response time of an M/M/1 queue with
  - Arrival rate λ = 0.7
  - Service rate μ = 1
  - Simulation ends when master clock is 16000s
- We repeat the experiment 30 times using different sets of random numbers
- For each independent experiments
  - We record the response time of all the jobs
  - Remove the transient part
  - Compute the mean response time using the steady state section
- We obtain 30 different estimates of the mean response time, one from each independent experiment
- These independent estimates allow us to find a confidence interval

Example (Cont'd)

- The blue circles show the estimated mean response time from the 30 independent experiments

- The red line is the 95% confidence interval

  - There is a 95% probability that the true mean response time that we want to estimate is in the interval [3.30,3.62]

- The green line is the theoretical mean response time (which you should not normally know).

Computing the confidence interval (1)

- Assume that you do *n* independent replications
- In each replication, you remove the transient part and compute an estimate of the mean steady state response time
  - Let us call your estimate from the kth replication, T(k)
- Compute the sample mean

$$\hat{T} = \frac{\sum_{i=1}^{n} T(i)}{n}$$

- And the sample standard deviation

$$\hat{S} = \sqrt{\frac{\sum_{i=1}^{n} (\hat{T} - T(i))^2}{n - 1}}$$

Note: for sample standard deviation, *(n-1)* is in the denominator, *not n*.

Computing confidence interval (2)

- There is a probability $(1-\alpha)$ that the mean response time that you want to estimate lies in the interval

$$[\hat{T} - t_{n-1,1-\frac{\alpha}{2}} \frac{\hat{S}}{\sqrt{n}}, \hat{T} + t_{n-1,1-\frac{\alpha}{2}} \frac{\hat{S}}{\sqrt{n}}]$$

where $t_{n-1,1-\frac{\alpha}{2}}$ is the upper $(1 - \frac{\alpha}{2})$
critical point for the Student t distribution
with $(n-1)$ degrees of freedom

Computing confidence interval (3)

- The value $t_{n-1, 1-\frac{\alpha}{2}}$ can be obtained from looking up

  the Student t distribution table
  - Note: A Student t table has been provided on the web site

- There are also programs that compute it
  - In Matlab, you can use tinv(1-alpha/2,n-1)

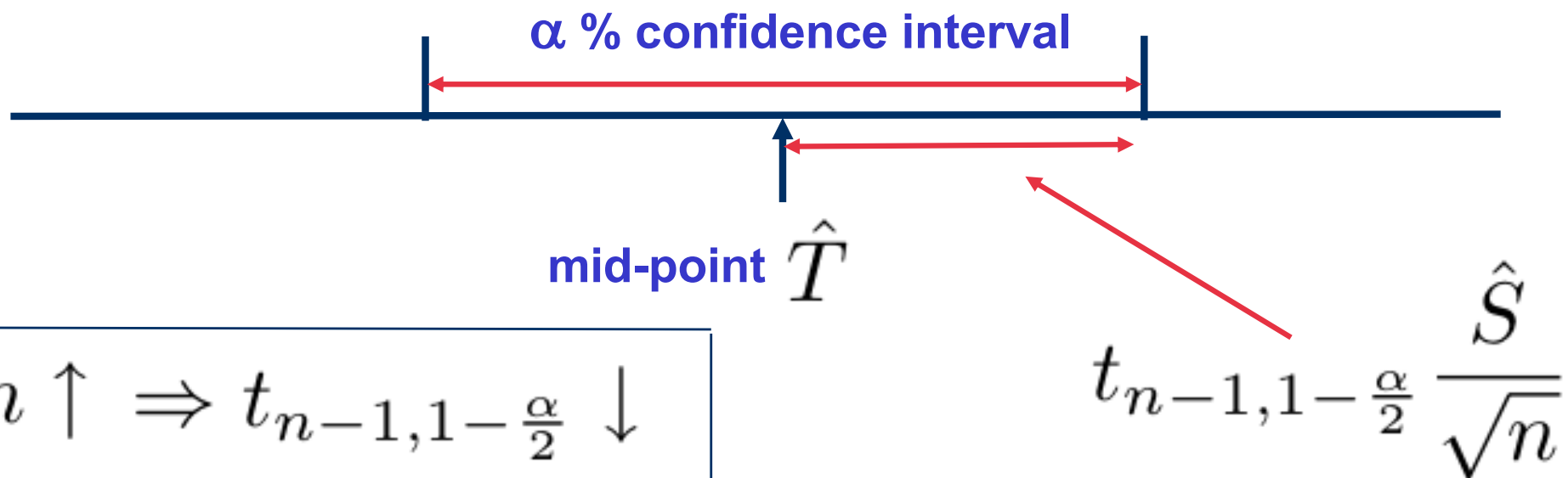# Example: Independent replications (cont'd) $t_{n-1,1-\frac{\alpha}{2}}$

- From the example on p.15
- The sample mean of (n =) 30 replications = 3.47
- The sample standard deviation of 30 replications is 0.43
- If we want to compute the 95% confidence interval, α = 0.05
- Since we did 30 independent experiments and want 95% confidence interval, we use $t_{29,0.975}$

- From the t-distribution table, the value of $t_{29,0.975}$ is 2.0452, the 95% confidence interval is

$$[3.47 - 2.0452\frac{0.43}{\sqrt{30}}, 3.47 + 2.0452\frac{0.43}{\sqrt{30}}] = [3.30, 3.62]$$

More on confidence interval

- Confidence interval

$$[\hat{T} - t_{n-1,1-\frac{\alpha}{2}} \frac{\hat{S}}{\sqrt{n}}, \hat{T} + t_{n-1,1-\frac{\alpha}{2}} \frac{\hat{S}}{\sqrt{n}}]$$

α % confidence interval

mid-point $\hat{T}$

$$n \uparrow \Rightarrow t_{n-1,1-\frac{\alpha}{2}} \downarrow$$

$$t_{n-1,1-\frac{\alpha}{2}} \frac{\hat{S}}{\sqrt{n}}$$

$$\Rightarrow t_{n-1,1-\frac{\alpha}{2}} \frac{\hat{S}}{\sqrt{n}} \downarrow$$

What can we get from simulation?

- If your queueing problem has a mathematical solution, you will get *one* value for the steady state mean response time

- If you simulate a queue to try to estimate the mean response time, you will *not* know the exact value of the steady state mean response time

- Simulation can only give you a confidence interval of what you want to estimate

- You can reduce the confidence interval by doing many independent replications!

Choice of simulation parameters (1)

- Simulation parameters
  - Length of simulation
  - Number of replications
  - Accuracy

- Unfortunately, there are *no* hard rules to choose them. You will need to do some trial and error
  - If the length of simulation is not long enough, you will need to increase it
  - If the number of replications is not enough to give you the desired accuracy, you will need to increase it

Choice of simulation parameters (2)

- Length of simulation
  - Must be longer than the transient
  - Should have a good number of data point in the steady state part
    - Hard to say what "good" is. Get a few hundred if you can. The more the better but of course your simulation will run longer

- Number of replications
  - You may want to have 5 replications to start with
  - After removing the transient, compute the confidence interval for your estimate.
  - Compare the width of your confidence interval with your desired accuracy. If the confidence interval that you have obtained is too wide, you will need to increase the number of replications.
  - Progressively (basically by trial-and-error), increase the number of replications until you get the desired level of accuracy

Comparing two systems: motivation

- An application of simulation is to compare two systems

- For example, in last week's revision question, you used simulation to compare the mean response time of
  - System 1: M/M/1 queue with $\lambda = 0.9$ and $\mu = 1$
  - System 2: M/M/2 queue with $\lambda = 0.9$ and $\mu = 0.5$ for both server

- If you use analytical method, you can find the steady state mean response time of both systems exactly and you compare two numbers

- If you use simulation, you get a confidence interval for each system instead. How do you compare them?

Example: Comparing two systems

- Let us assume our goal is to use simulation to compare:
  - System 1: M/M/1 queue with $\lambda = 0.9$ and $\mu = 1$
  - System 2: M/M/2 queue with $\lambda = 0.9$ and $\mu = 0.5$ for both server

- For each system we carry out 3 independent replications
  - That is, we use 6 sets of independent random numbers together

- After removing the transient, the estimated mean response times are:
  - System 1: 6.8769, 8.5769, 10.6340
  - System 2: 8.8087, 7.4616, 9.1565

- In order to compare them, let us pair up these results
  - 1st experiment for System 1 with 1st experiment for System 2
  - 2nd experiment for System 1 with 2nd experiment for System 2 etc.

A paired-*t* confidence interval

- Let us summarise the data in a table
  - EMRT = estimated mean response time

	EMRT System 1	EMRT System 2	EMRT System 2 - EMRT System 1
Rep. 1	6.8769	8.8087	1.9318
Rep. 2	8.5769	7.4616	-1.1154
Rep. 3	10.6340	9.1565	-1.4775

- We compute the 100 (1-α)% confidence interval of the difference between 2 systems (= last column)

- Let us denote the computed confidence interval by [p,q]
  - Case 1: p,q > 0 ➜ System 1 is better than System 2 with probability (1-α)
  - Case 2: p,q < 0 ➜ System 2 is better than System 1 with probability (1-α)
  - Case 3: q > 0 & p < 0 ➜ Systems 1 and 2 are not different with probability (1-α)

Example: Paired-*t* confidence interval

- We compute the 95% confidence interval of the data showed in the last slide, the confidence interval is:
  - [-4.8721,4.4314]

- Therefore, with 95% probability that the mean response times of the two systems are not different

- Hmmm, we have a problem here, we know from queueing theory that System 1 has a better mean response time than System 2, but our simulation does not seem to be able to distinguish them.

- What can we do?
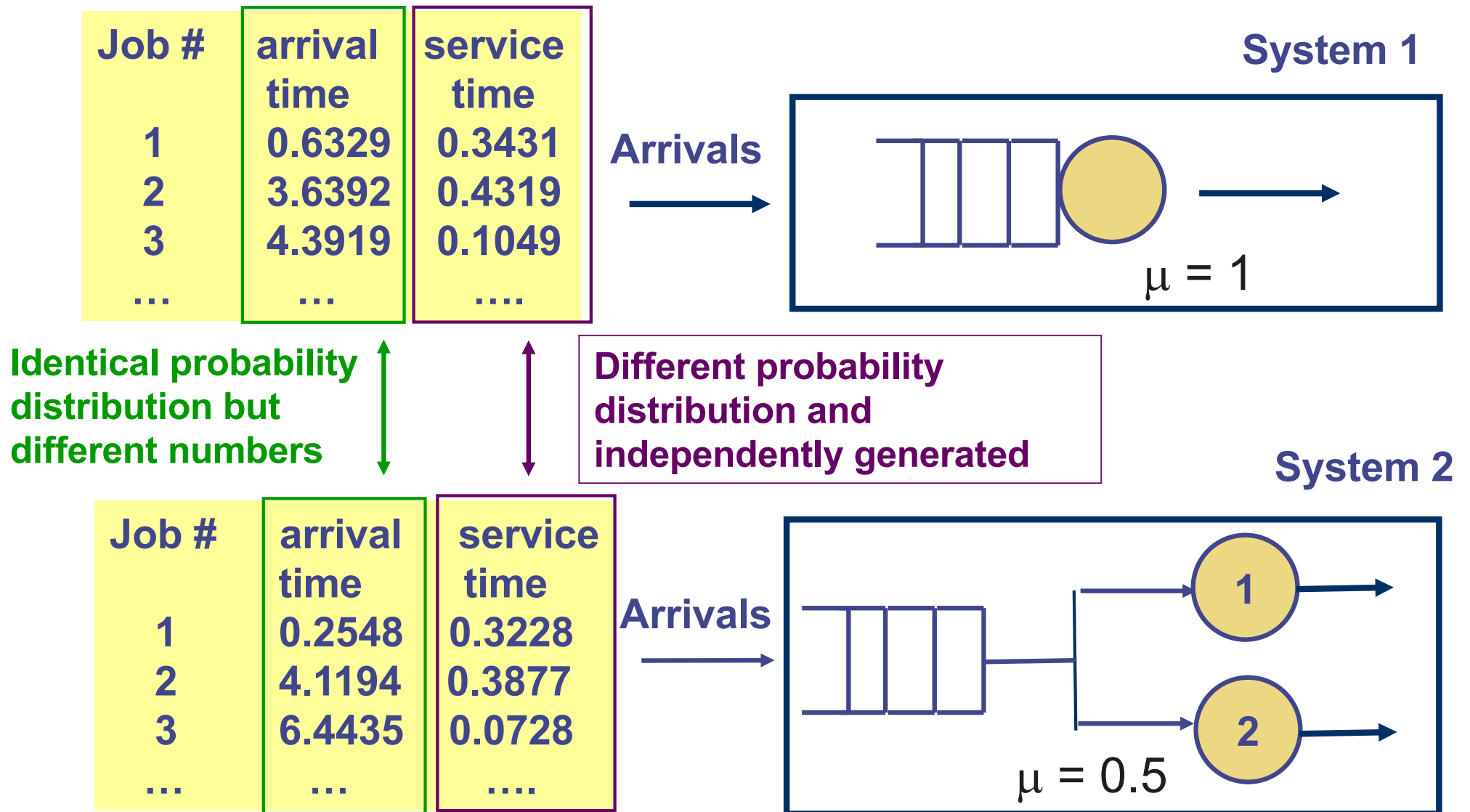
Let us increase the number of replications

- Since increasing the number of replications can reduce the width of the confidence interval, let us try that.
- Let us try 5, 10, 20, 30 replications

# independent replications	95% Confidence interval of EMRT System 2 - EMRT System 1
5	[-4.9540, 5.0242]
10	[-1.5347, 2.8020]
20	[-1.2724, 1.9870]
30	[-0.6001, 1.8046]

- Increasing the number of replications does reduce the width of the confidence interval
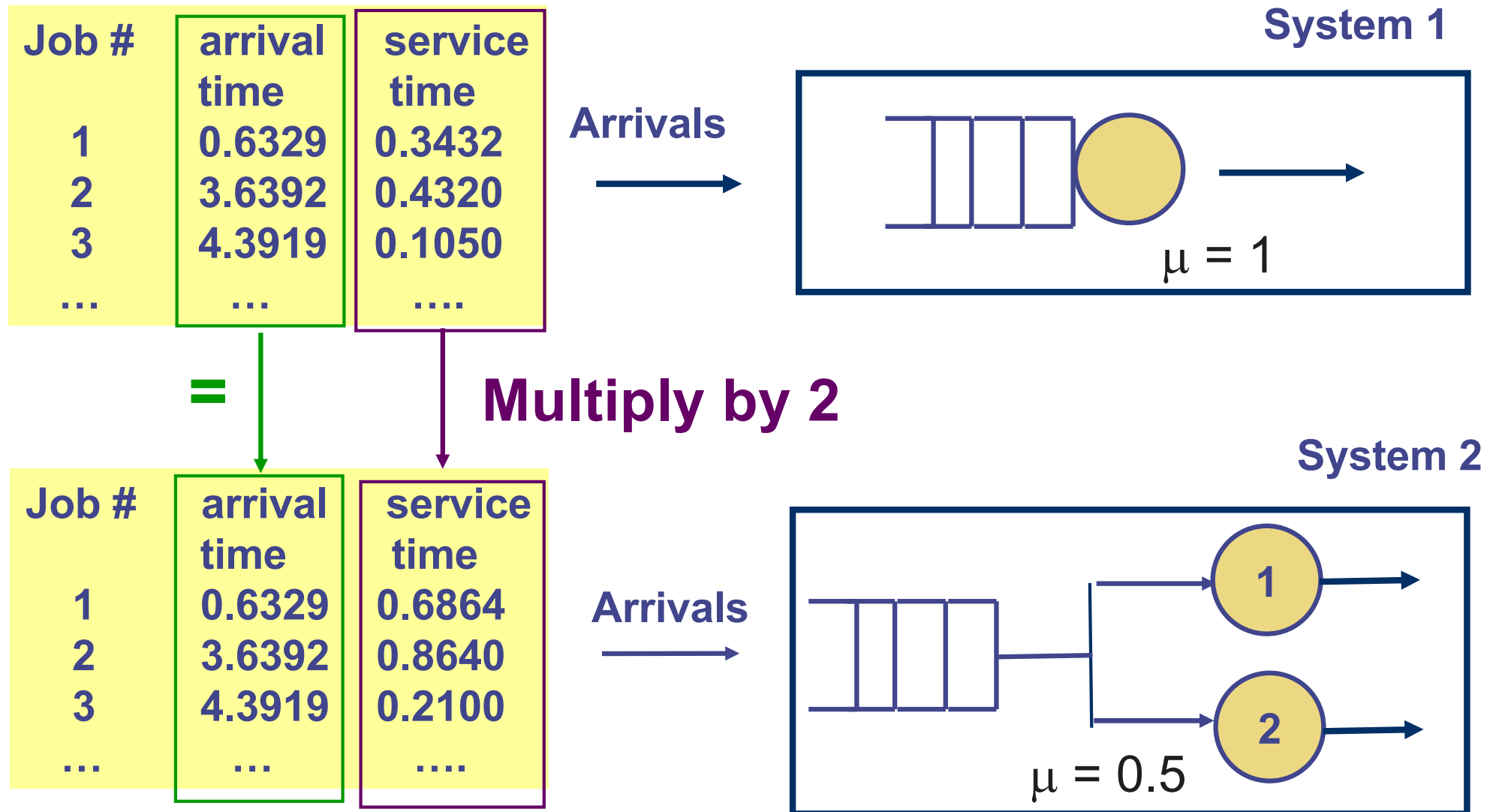- However, we still cannot conclude which system is better

Let us have a look at how we did our experiments …

- We did our experiment with independent random numbers

Job #	arrival time	service time
1	0.6329	0.3431
2	3.6392	0.4319
3	4.3919	0.1049
…	…	….

System 1

Arrivals

$\mu = 1$

Identical probability distribution but different numbers

Different probability distribution and independently generated

System 2

Job #	arrival time	service time
1	0.2548	0.3228
2	4.1194	0.3877
3	6.4435	0.0728
...	…	….

Arrivals

1

2

$\mu = 0.5$

Common random numbers method

- An alternative is to compare two systems under similar condition

Job #	arrival time	service time
1	0.6329	0.3432
2	3.6392	0.4320
3	4.3919	0.1050
…	…	….

System 1

Arrivals

$\mu = 1$

=

Multiply by 2

Job #	arrival time	service time
1	0.6329	0.6864
2	3.6392	0.8640
3	4.3919	0.2100
…	…	….

System 2

Arrivals

1

2

$\mu = 0.5$

Common random numbers method

- A method to reduce the variance when comparing two alternative systems is to subject them to similar experimental condition

- In each replication, generate only one arrival time and one service time sequence
  - Apply this to both systems
  - Note: Service time may need to be adjusted according to service rate

- In next replication, generate a new arrival time and a new service time sequence
  - Apply this to both systems

- This method can reduce the variance if the behaviour of the two systems is positively correlated

Applying common random numbers to our problem (1)

- Let us apply the common random numbers method to compare
    - System 1: M/M/1 queue with λ = 0.9 and μ = 1
    - System 2: M/M/2 queue with λ = 0.9 and μ = 0.5 for both server

- Let us carry out 5 replications

- In each replication, we generate one arrival time sequence and one service time sequence (adjusted by service rate) and apply to both systems

Applying common random numbers to our problem (2)

- Let us compare the estimated mean response time (EMRT) from the 5 replications:

	EMRT System 1	EMRT System 2	EMRT System 2 - EMRT System 1
Rep. 1	8.3022	8.8087	0.5065
Rep. 2	6.8809	7.4616	0.5807
Rep. 3	8.5769	9.1565	0.5796
Rep. 4	10.6340	11.3409	0.7069
Rep. 5	16.2648	16.6485	0.3837

- Observation: The EMRT of System 2 is higher than that of System 1 in all 5 replications
- If we compute the 95% confidence interval of the last column, we get [0.4046,0.6983]
- There is a 95% probability that System 1 is better than System 2

Comparing two methods

- Let us compare using common random number (CNR) method or not

# independent replications	95% Confidence interval of EMRT System 2 - EMRT System 1	
	Not using CNR	Using CNR
5	[-4.9540, 5.0242]	[0.4046, 0.6983]
10	[-1.5347, 2.8020]	[0.4705, 0.6103]
20	[-1.2724, 1.9870]	[0.5127, 0.5942]
30	[-0.6001, 1.8046]	[0.5026, 0.5786]

- Observations
  - By using CNR, all 95% confidence interval does not include 0
  - The width of the confidence interval for CNR method is a lot lower!
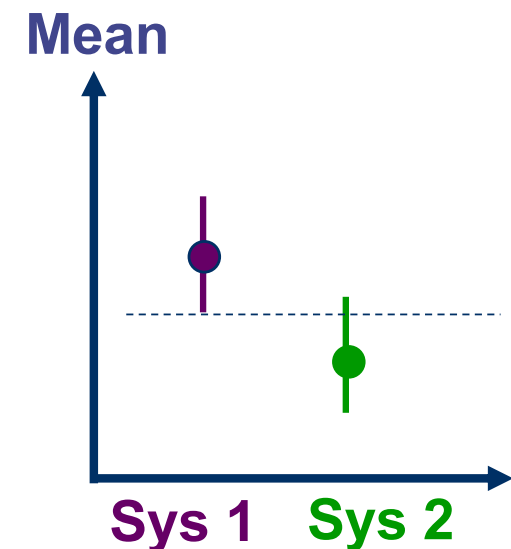
Approximate visual test

- Let us assume that you know the mean response time and its confidence interval (CI) for 2 systems: **System 1** and **System 2**

- Consider the following 3 possibilities:
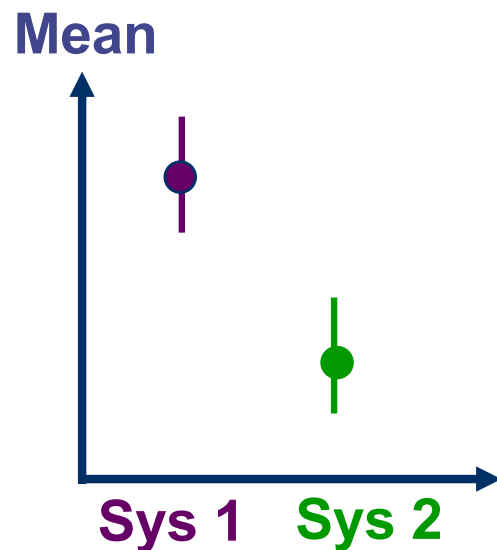

CIs do not overlap Mean of System 1 > Mean of Sys. 2
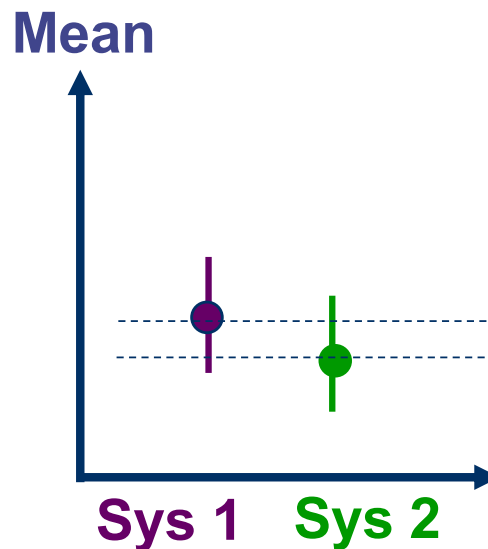
CIs overlap and mean of a system is in the CI of the other: System are *not* different

CIs overlap and mean of any one is not in the CI of the other: do *t*-test

Ex: Multicast protocol design for wireless mesh networks

- Comparing 3 multicast protocols (WCMA, **SPT** and **RCAM**) for wireless mesh networks
- The thin vertical line shows the confidence interval
- What conclusion can you draw?
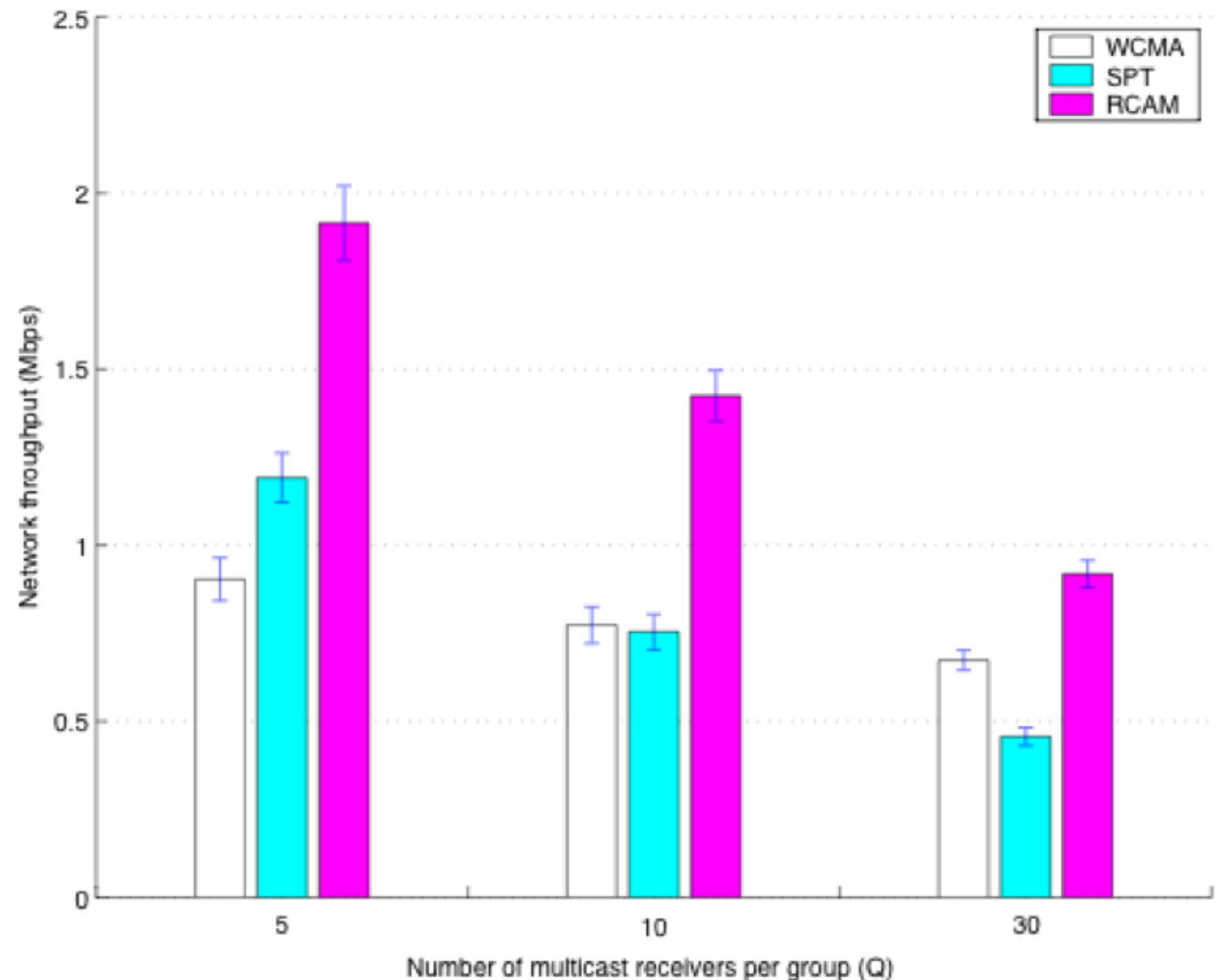
Simulation tools and some applications (1)

- You do not always have to write your own simulation programs from scratch
- There are plenty of simulation tools available
  - Many with GUI

- Simulation tools are used in a lot in computer networking research
  - Protocol #1 is the existing protocol, you have designed Protocol #2. You want to see whether Protocol #2 is better or not.

  - You have two options (Option #1 and Option #2) to design a network. Which option is better?

Simulation tools and some applications (2)

- Some examples of publicly available simulation tools
  - General purpose: OMNet++
    - http://www.omnetpp.org/
  - For networking research: ns3
    - http://www.isi.edu/nsnam/ns/
- Some commercial tools
  - For network design: OPNET, Qualnet
    - http://www.opnet.com/
    - http://web.scalable-networks.com/content/qualnet

- **Important note: These tools save you time in writing simulation program but don't forget that you still need to analyse your simulation results using statistically sound methods!**

Summary

- Simulation is not just a computer programming exercise
- You need to make sure that your program is correct
- It is also important to analyse your results statistically
- Methods discussed include
  - Transient removal technique
  - Confidence interval
  - Determining number of replications
  - Comparing 2 alternatives
  - Variance reduction
- Unfortunately, a lot of published research papers in computer networking did not do sound statistical analysis
  - Optional reading: Pawlikowski et al, "On credibility of simulation studies of telecommunication networks", IEEE Communications Magazine, Pages 132-139, January 2002.

References

- The primary reference is Law and Kelton, "Simulation Modelling and Analysis"
    - Transient removal, Sections 9.1, 9.2 and 9.5
    - Replication method, Section 9.5.2
    - Comparing two alternatives, Section 10.1, 10.2 (10.2.1 only)
    - Common random numbers, Section 11.2
- Raj Jain, "The Art of Computer Systems Performance Analysis" has materials on
    - Transient removal methods, Section 25.3
    - Calculating confidence interval, Section 13.2
    - Comparing two alternatives, Sections 13.3, 13.4 (13.4.1 and 13.4.3 only)
- Note that we have only touched on the basic of statistical analysis of simulation data. The above two books (outside the specified sections) will provide you with more in depth discussion on the topic.
- If you are interested to know the mathematical background on confidence interval, student t-distribution etc., a possible reference is Wackerly et al, "Mathematical Statistics with Applications".