COMP9334
Capacity Planning for Computer Systems and Networks

Week 5A Simulation (2). Independent replications. Confidence interval.

Week 4B

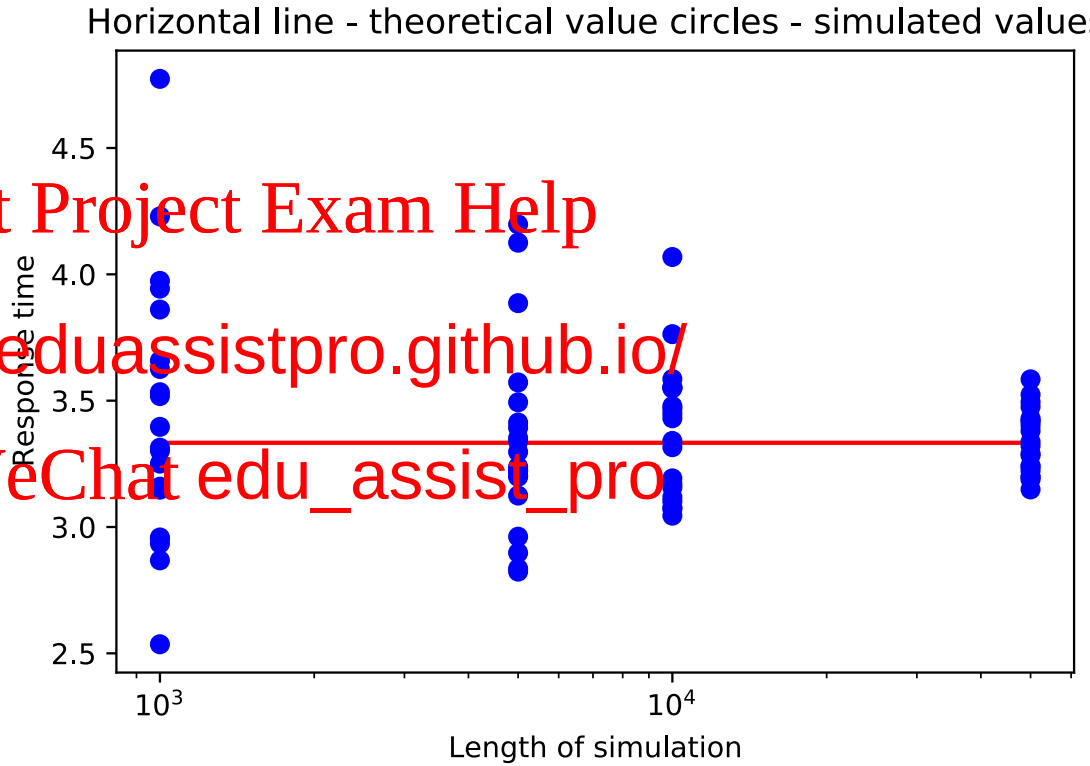
- Two topics
  - How to structure discrete event simulation of queues
  - How to use the Python random library to generate random numbers for inter-arrival and service times

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

- However, there are a number of problems …

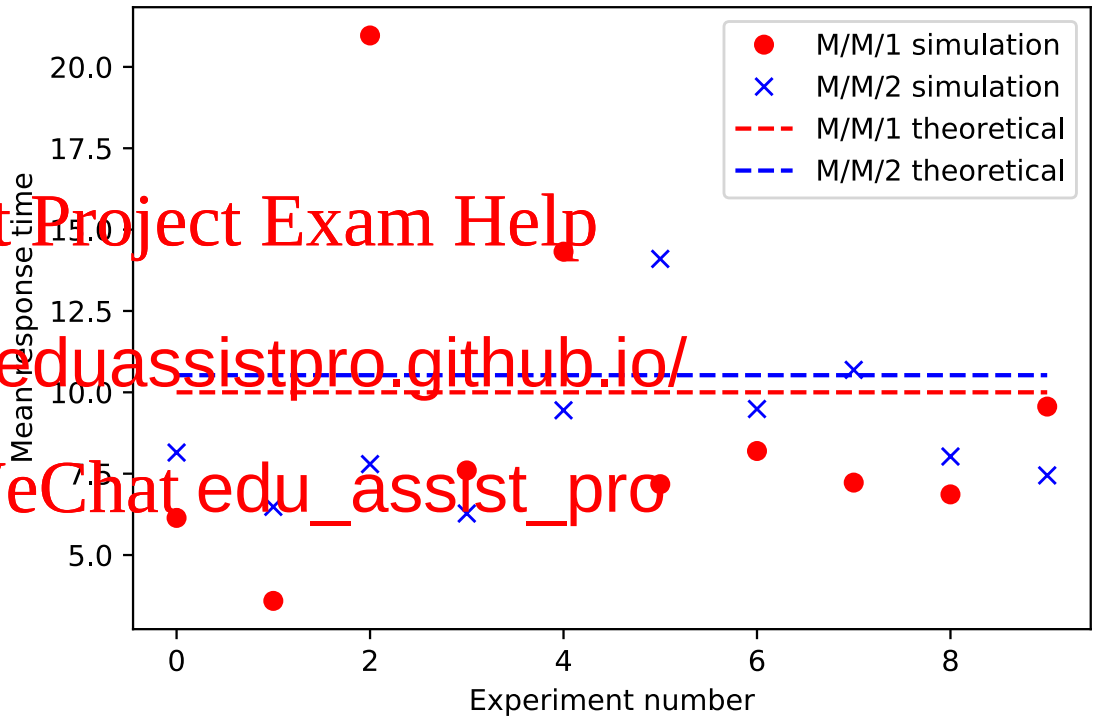
Problem: data interpretation, simulation length

- Week 4B's revision problem #1. The problem asks you to:
  - use 4 different simulation end times: 1000, 5000, 10000, 50000
  - For each end time, do 20 runs
- Figure on the right sh the solution
  - Each dot is the average response time for each run
  - The red horizontal line shows the theoretical
- What observations can you make?



Problem: How do we compare 2 alternative choices?

- Week 4B's Revision Problem #2. The question asks you to simulate each of the following 2 queues 10 times:
  - M/M/1 queue with λ = 0.9 and μ = 1
  - M/M/2 queue with λ = μ = 0.5
- From Queueing theory, we expect the M/M/1 system to have a lower mean response time but do the simulation results suggest that?



Analysis of simulation results

A very important topic but it is very often ignored

Simulation is an empty about

Writing a computer program
Verifying the correctness of the program
ning the simulation once and present
results

Verifying the correctness of the s am 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?
- Confidence interval

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 the *steady state mean response*

- Recall that we talked about
  - Transient and steady state behaviour of queue in Week 2B
  - Steady state of Markov chain in Week 3B

What is steady state? (1)

- Let us simulate an M/M/1 queue with
  - Arrival rate λ = 0.7
  - Service rate μ = 1
  - Simulation ends when master clock is 50000s
- In this simulation, we record the response time for each job
  - Let X(k) = Res
  - The next page
- Let N denote the number of jo simulation
  - N = 35000 for our simulation
- In Week 5A, we computed the mean response time using

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

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

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

- For example, if k = 5, then

- 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 data to compute the steady state value
- We should remove the transient 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 Revision Problem 5A

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

- Note: We used too simple a met pute the mean in Week 4B but I didn't want to complic 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.**

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 m e interval [3.1,3.3].
  - We call the int *the confidence interval*.
- Independent replications: Rep ulation 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 $\lambda = 0.7$
  - Service rate $\mu = 1$
  - Simulation ends when master clock is 16000s
- We repeat the experiment 30 times using different sets of random number
- For each indepe
  - We record the response time
  - 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% pro 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 k^{th} replication T(k)
- Compute the sa

- 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*.
See also the note on p.21.

Computing confidence interval (2)

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

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

with $(n-1)$degrees of freedom

- If α is 0.05, then it means there is a 0.95 probability that the mean response time is in the calculated confidence interval

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 pit

  - In Python

  - In Matlab, you can use tinv(1-alpha/2,n-1)

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

- Five replications with mean response times:
  - 0.31, 0.37, 0.34, 0.36, 0.39
- The sample mean of (n =) 5 replications = 0.354
- The sample standard deviation of 5 replications is 0.0305
- If we want to compute the 95% confidence interval, α = 0.05
  - Since we did 5 in nd want 95% confidence interval, we use $t_{4,0.975}$
- From the t-distribution table, the value of $t_{4,0.975}$ is 2.776, the 95% confidence interval is

$$\left[0.354 - 2.776\frac{0.0305}{\sqrt{5}}, 0.354 - 2.776\frac{0.0305}{\sqrt{5}}\right] = [0.316, 0.392]$$

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

Sample program and caution

- The sample program comp_confidence_interval.py shows you how you can do the calculations on the previous slide using Python libraries numpy and scipy.stats

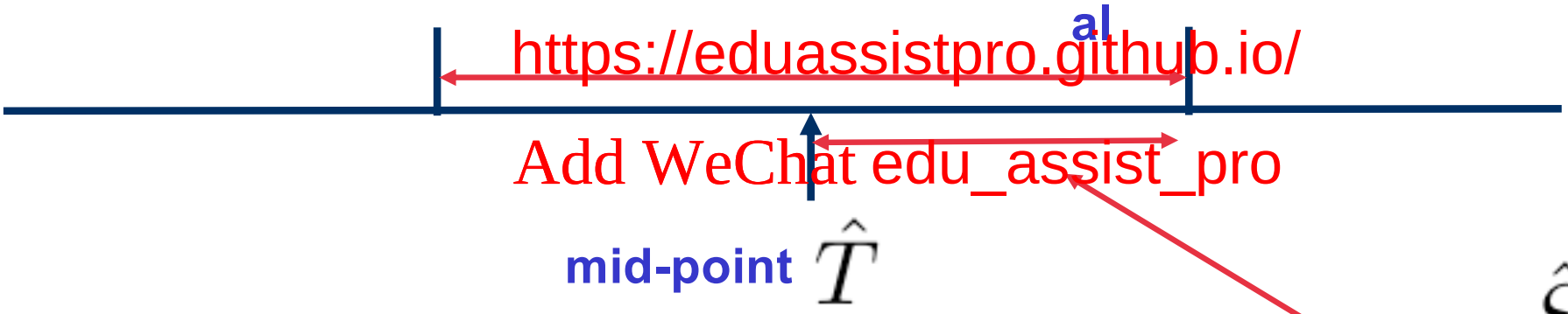
- Caution: We need to compute sample standard deviation

which means you need to divide by (n-1), not divide by n. Some Python functions divide by (n-1) by default but some others need to use an option. See the sample program. If in doubt, use a simple example to check.

More on confidence interval

- Confidence interval

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



What happens if you make n bigger, i.e. do more independent replications ?

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 know the exact value of the steady state me
- Simulation can only give you a _ 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, th to choose them. You will need to do s

  - If the length of simulation is not l 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 replic

  - You may want to have 5 replic rt with

  - After removing the transient, c 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

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 int
  - Determining n

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
- Raj Jain, “The Art of Computer Systems Performance Analysis” has materials on
  - Transient removal methods, Section 25.3
  - Calculating confi
- If you are intere matical background on confidence interval studen tion etc., a possible reference is Wackerly et al, “ al Statistics with Applications”.