TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Business Process Simulation

Lecture 5

Department of Industrial Engineering and Innovation Sciences, Information Systems Group

Overview on lecture modules

a) Tutorial session
b) Pilot simulation runs
c) Simulation output analysis

Overview of today's lecture

STEP 4: Executable Model

- Recap
- Pilot simulation runs

STEP 6: Experiments and Output Analysis

- Introduction to simulation experiments
- Output analysis
- Simulation parameters

Recap STEP 4: Simulation Methodology (7 steps)

Thinking phase

STEP 1: Project definition

STEP 2: Design the simulation study

STEP 3: Conceptual model

Execution phase

STEP 4: Executable model and verification

STEP 5: Validation

STEP 6: Experiments and output analysis

STEP 7: Conclusion

TU/e

Recap STEP 4: Simulation Methodology (7 steps)

STEP 1: Project definition

STEP 2: Design the simulation study

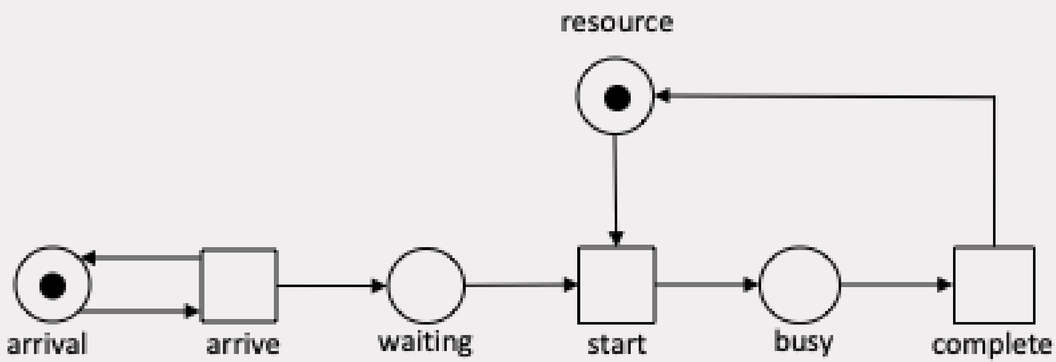
STEP 3: Conceptual model

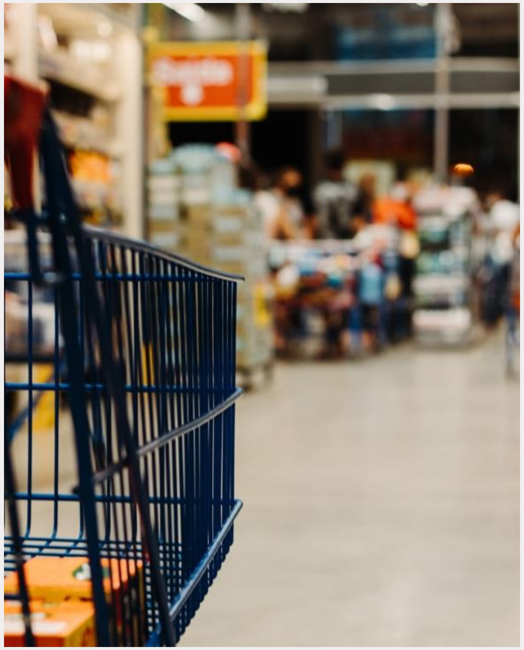
STEP 4: Executable model and verification

STEP 5: Validation

- **Dependent on simulation tool (SimPN)**
- **Verifcation: does the model behave as intended/expected?**

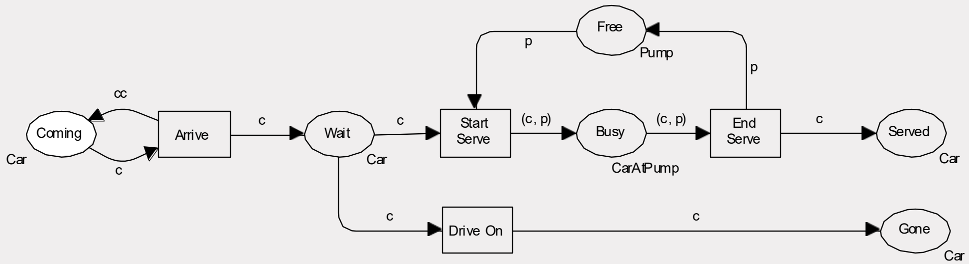
Introduction to SimPn





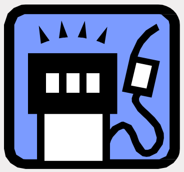
```
from simpn.simulator import SimProblem, SimToken

shop = SimProblem()

resources = shop.add_var("resources")
customers = shop.add_var("customers")

def process(customer, resource):
    return [SimToken(resource, delay=0.75)]

shop.add_event([customers, resources], [resources], process)

resources.put("cassier")
customers.put("c1")
customers.put("c2")
customers.put("c3")

from simpn.reporters import SimpleReporter

shop.simulate(10, SimpleReporter())
```

Recap STEP 4: Mapping process model to SimPN



Process model	SimPn	Function
(Coming) + Arrive	Start_event start	Create car instances
Wait	to_check_queue, waiting	Variables storing the tokens generated by the arrival event and the tokens admitted to the pupm queue
Wait+(Start Serve OR Drive On)	Check_queue_length	Checks whether the car is admitted to the station
Served, Gone	Done, to_leave, End_event leave, End_event complete	Variables storing served and not served tokens, together with the corresponding end events
Free	pump	Variable storing the resources
Start serve + Busy + End Serve +Free	Task refueling	Task serving the cars at the station

Recap STEP 4: Solution patterns in SimPN



Pattern 1a: Balking: deciding to not join the queue
Petrol Station example

Pattern 1b: Jockeying: switching queues
Checkout example, Harbour example

Pattern 1c: Reneging pattern: leaving queue after certain time
Car Wash example

Pattern 2: Sharing resources with other activities
Coffee Break example

Pattern 3: Shifting task priorities
Order Processing example

Simulation output

Three ways to gain output information

1. Simple reporting

```
([(arrival, 1@0)], 0, arrive)
([(waiting, c1@0), (resource, r1@0)], 0, start)
([(busy, ('c1', 'r1')@0.8)], 0.8, complete)
([(arrival, 2@1)], 1, arrive)
([(waiting, c2@1), (resource, r1@0.8)], 1, start)
([(arrival, 3@1.7)], 1.7, arrive)
([(busy, ('c2', 'r1')@1.9)], 1.9, complete)
([(waiting, c3@1.7), (resource, r1@1.9)], 1.9, start)
([(busy, ('c3', 'r1')@2.6)], 2.6, complete)
```

Simulation output

Three ways to gain output information

- Simple reporting
- Process reporter
  - Per case: average waiting time, average processing time, average cycle time
  - Per resource: utilization rate

Simulation output

Three ways to gain output information

- Simple reporting
- Process reporter
- Event log reporter

```
case_id,task,resource,start_time,completion_time
arrive0,arrive,,2020-01-01 00:00:00.000000,2020-01-01 00:00:00.000000
arrive1,arrive,,2020-01-01 00:05:42.125312,2020-01-01 00:05:42.125312
arrive0,answer_call,e1,2020-01-01 00:00:00.000000,2020-01-01 00:13:05.878894
arrive0,complete,,2020-01-01 00:13:05.878894,2020-01-01 00:13:05.878894
arrive1,answer_call,e1,2020-01-01 00:13:05.878894,2020-01-01 00:23:50.859557
arrive1,complete,,2020-01-01 00:23:50.859557,2020-01-01 00:23:50.859557
arrive2,arrive,,2020-01-01 00:34:38.307065,2020-01-01 00:34:38.307065
arrive3,arrive,,2020-01-01 00:35:46.197993,2020-01-01 00:35:46.197993
arrive4,arrive,,2020-01-01 00:38:17.643895,2020-01-01 00:38:17.643895
arrive2,answer_call,e1,2020-01-01 00:34:38.307065,2020-01-01 00:46:28.904668
arrive2,complete,,2020-01-01 00:46:28.904668,2020-01-01 00:46:28.904668
arrive5,arrive,,2020-01-01 00:56:08.076029,2020-01-01 00:56:08.076029
arrive3,answer_call,e1,2020-01-01 00:46:28.904668,2020-01-01 00:58:14.238229
arrive3,complete,,2020-01-01 00:58:14.238229,2020-01-01 00:58:14.238229
arrive6,arrive,,2020-01-01 00:58:17.973171,2020-01-01 00:58:17.973171
arrive4,answer_call,e1,2020-01-01 00:58:14.238229,2020-01-01 01:10:04.181595
```

Simulation output

Three ways to gain output information

- Simple reporting
- Process reporter
- Event log reporter
- Your own reporter function

Simulation output

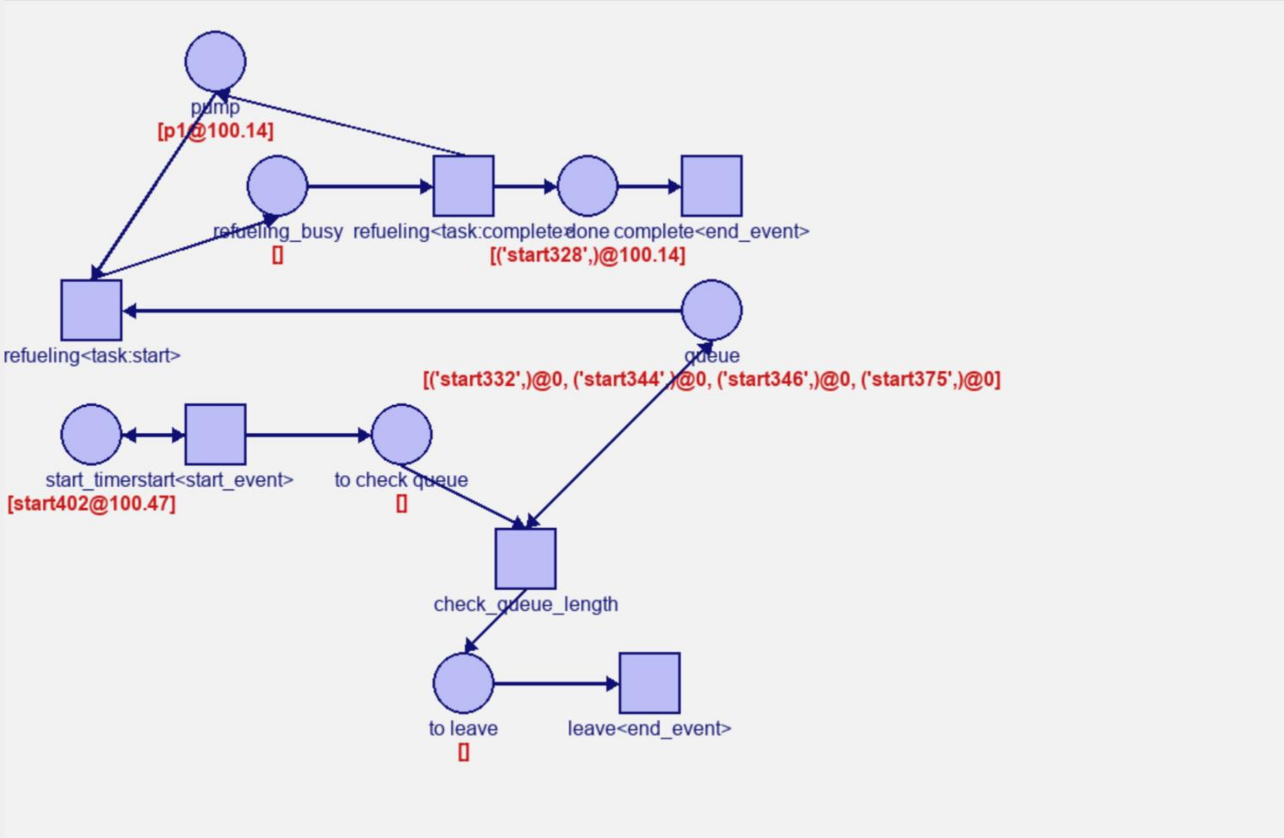
Three ways to gain output information

- Simple reporting
- Process reporter
- Event log reporter
- Your own reporter function

⇒ Pilot simulation runs, verification, and validation

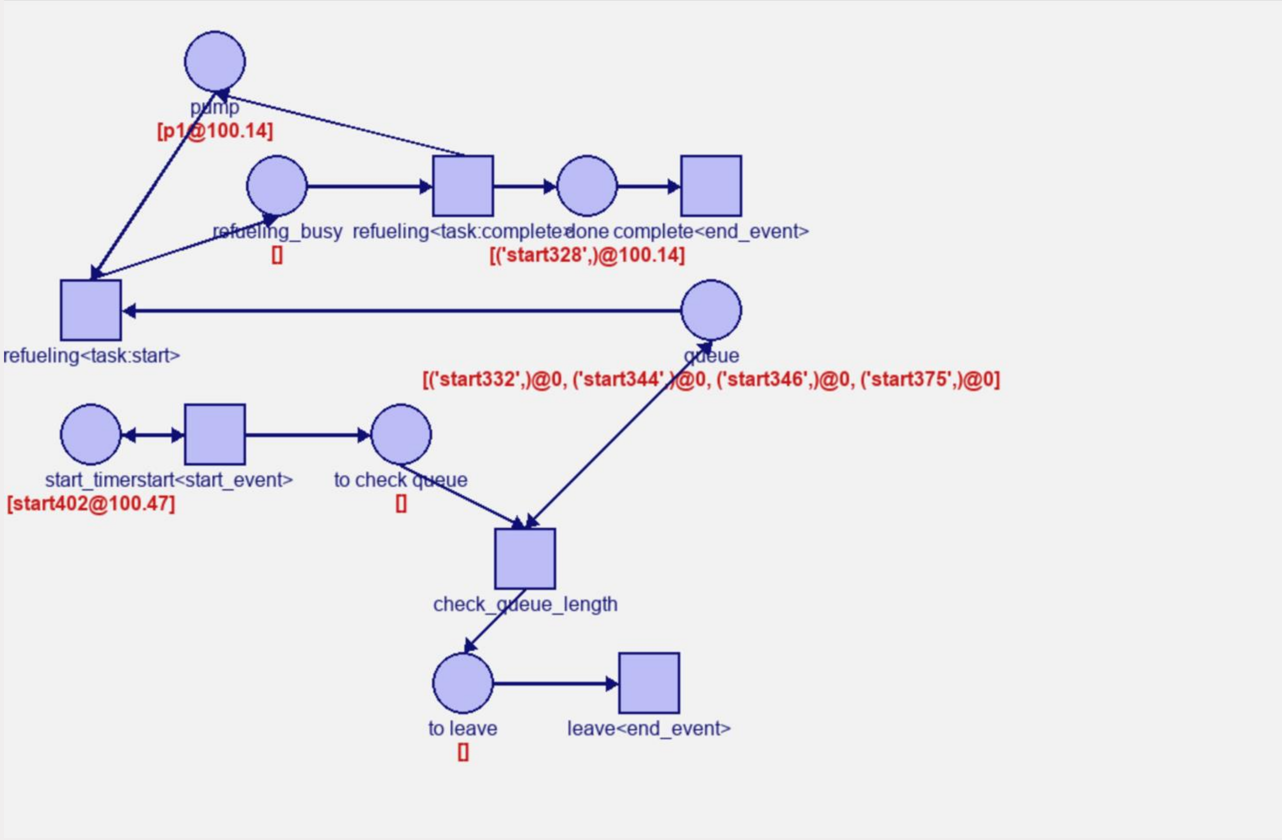
Business Process Simulation

5a) Pilot simulation runs

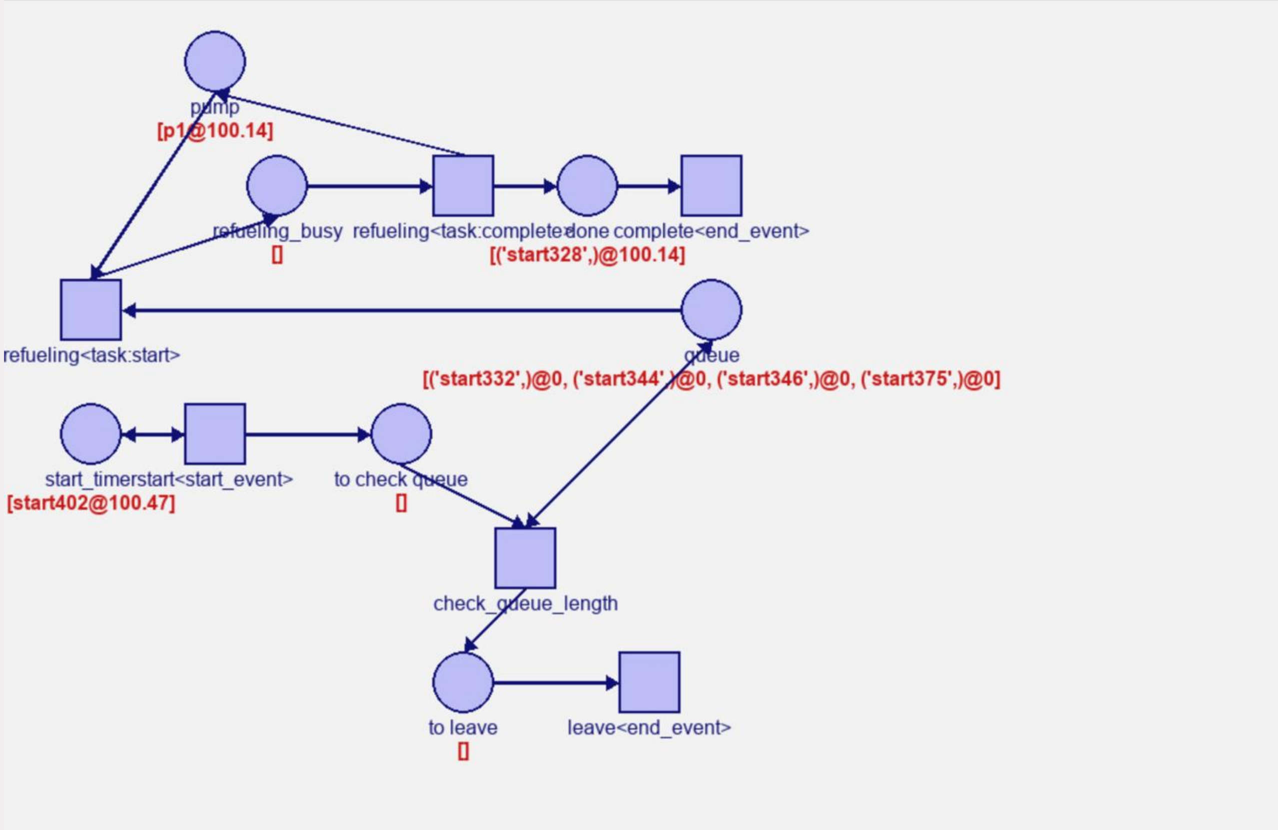


- Percentage cars gone: 18/252 = 7.14%
- Based on one pilot run

A pilot run is a simulation run of reasonable length (without having determined the simulation parameters yet)

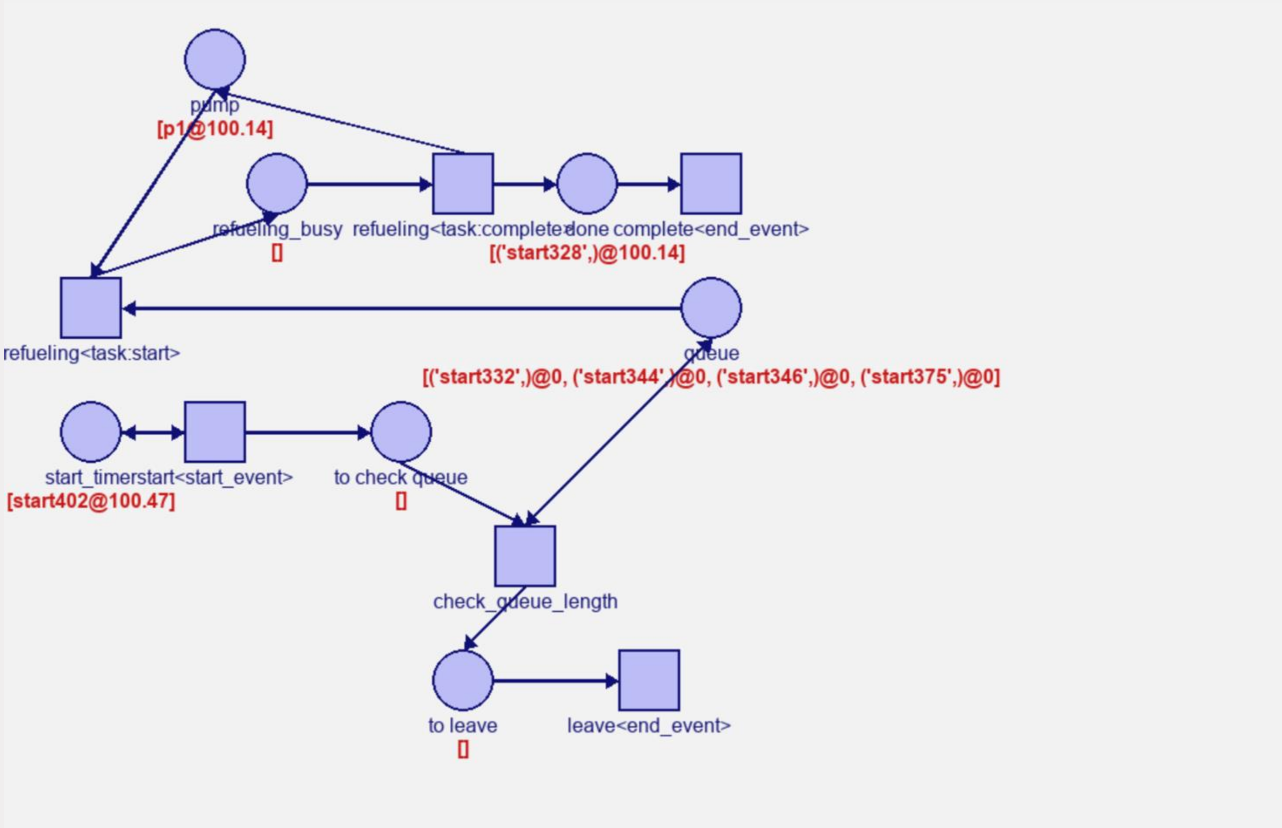


Throughput time	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n
Replication 1	5.56	8.41	3.95	…	10.21	9.68

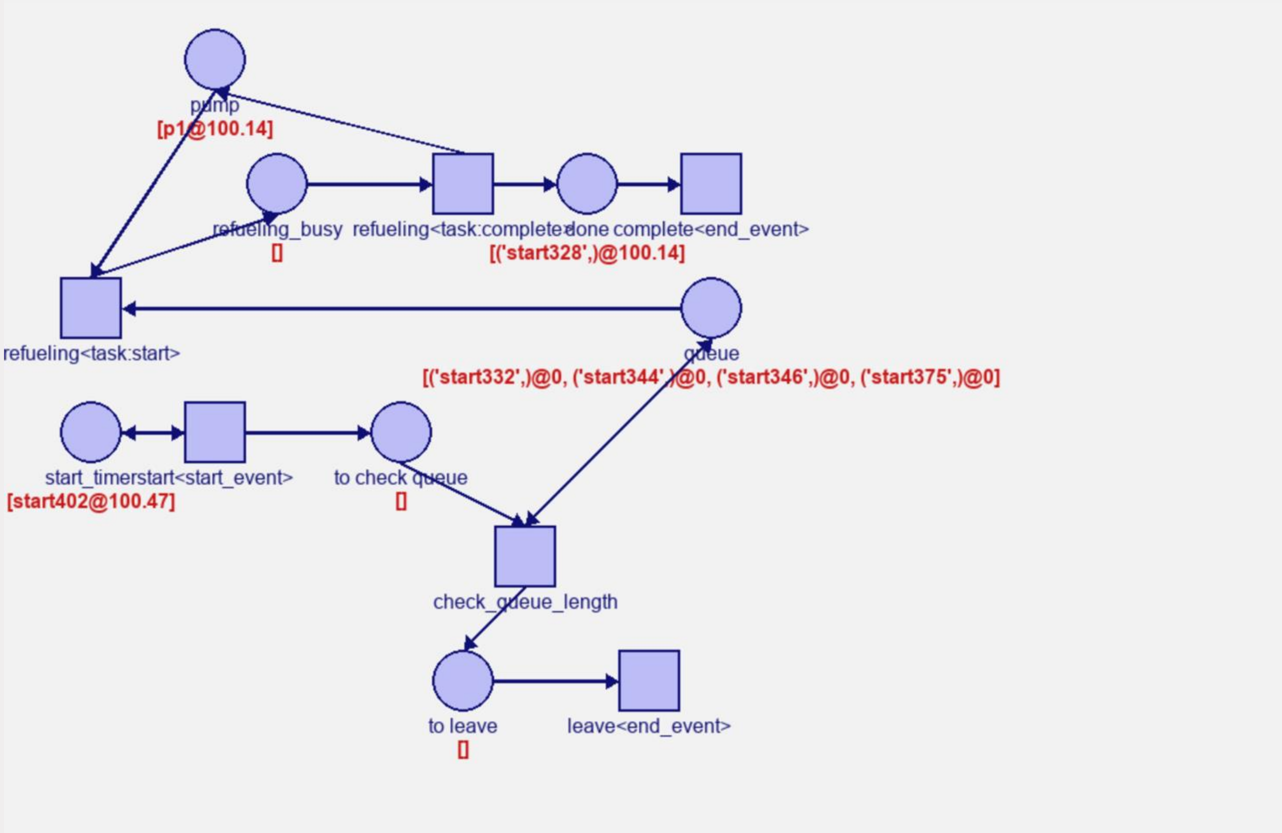


Throughput time	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n
Replication 1	5.56	8.41	3.95	…	10.21	9.68
Replication 2	3.18	6.03	8.67		7.25	8.44

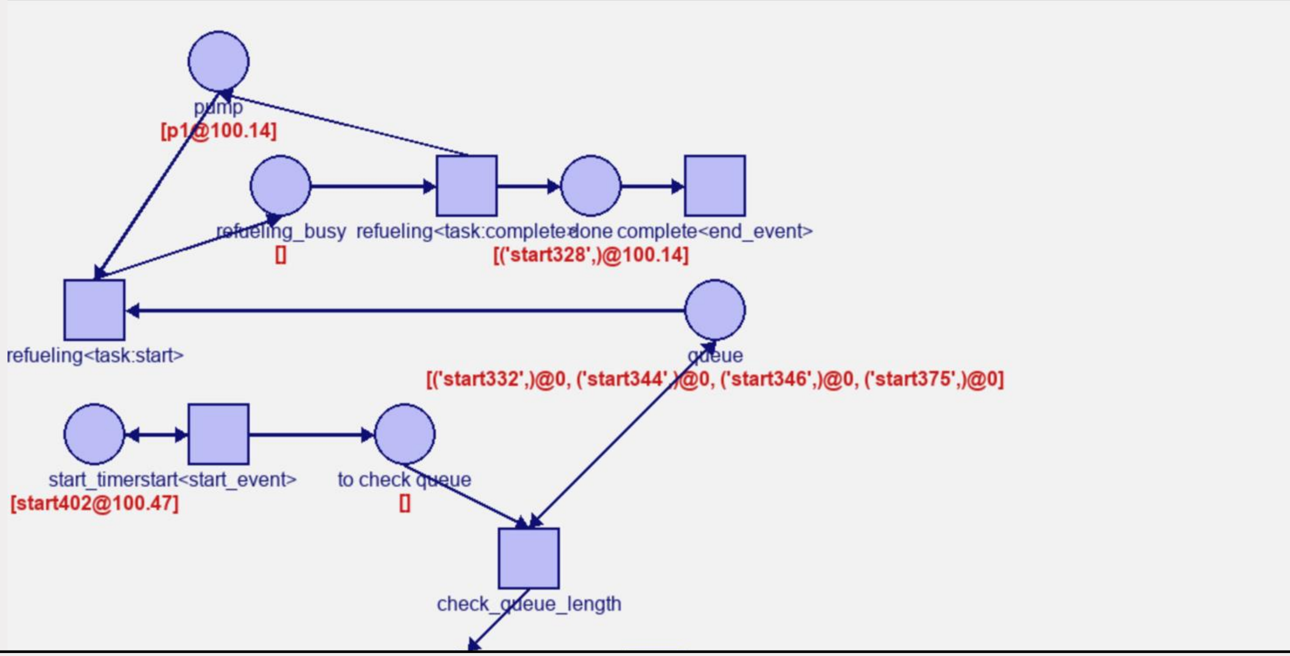
TU/e



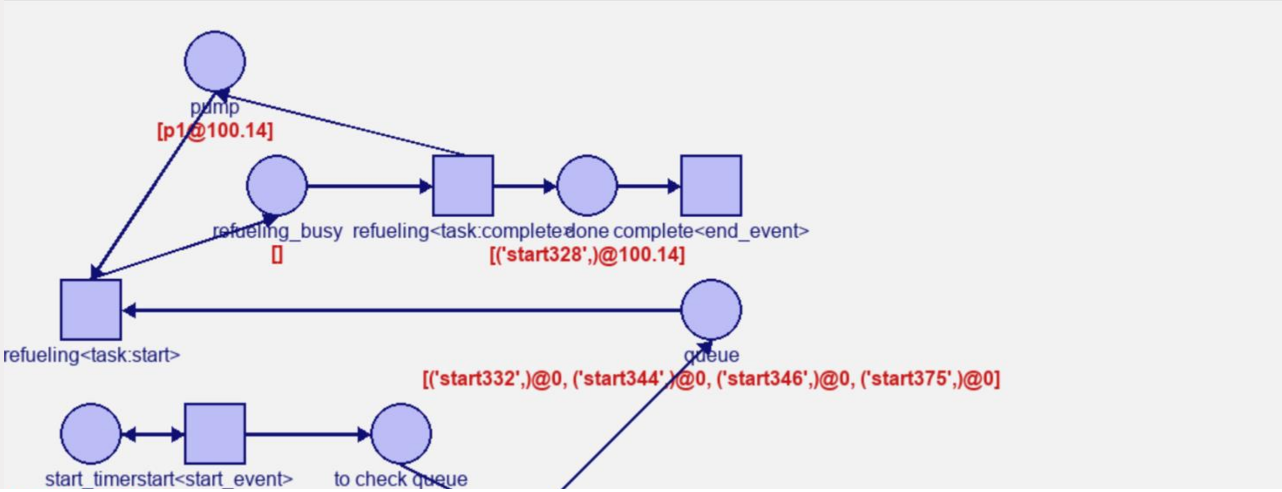
Throughput time	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n
Replication 1	5.56	8.41	3.95	…	10.21	9.68
Replication 2	3.18	6.03	8.67		7.25	8.44
Replication 3	4.76	5.62	5.78	…	10.36	11.98

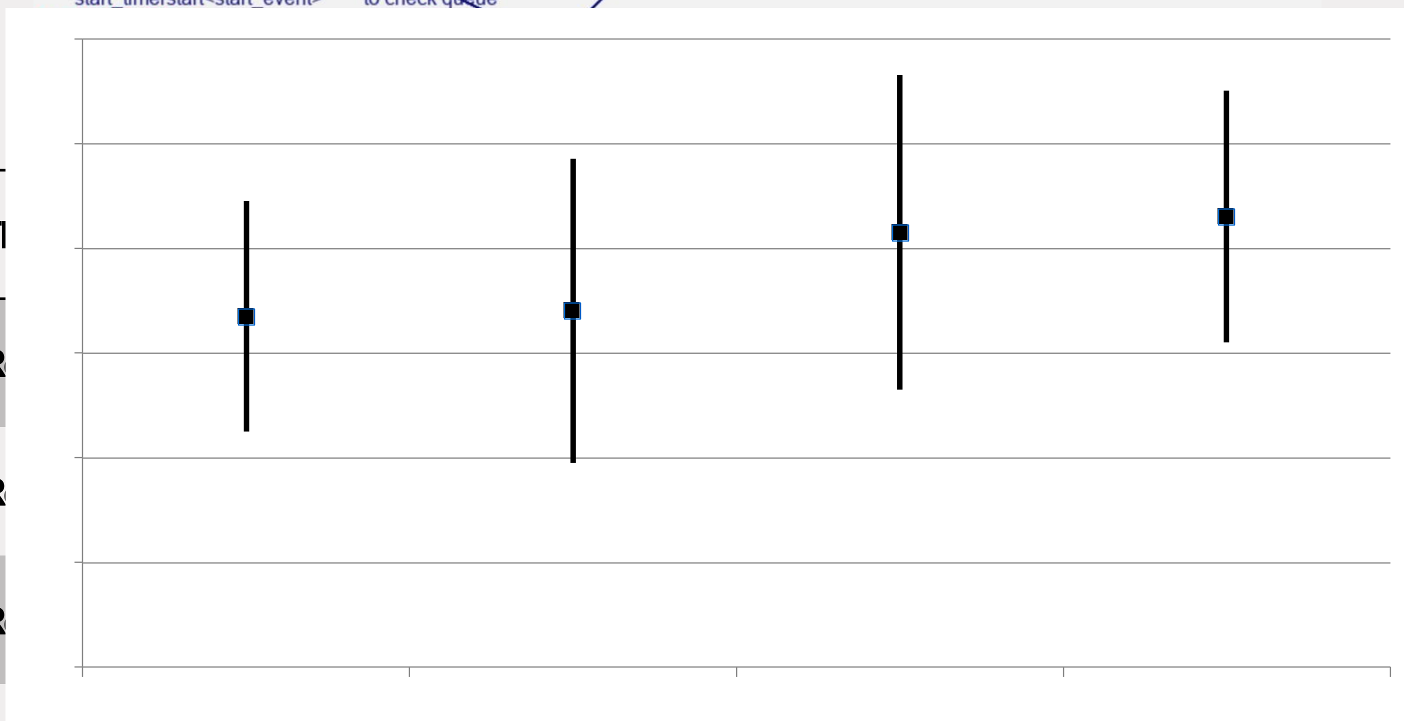


Throughput time	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	5.56	8.41	3.95	…	10.21	9.68	**7.47** (± 0.22)
Replication 2	3.18	6.03	8.67		7.25	8.44	**7.48** (± 0.29)
Replication 3	4.76	5.62	5.78	…	10.36	11.98	**7.63** (± 0.30)



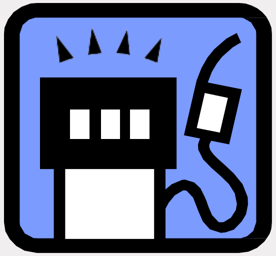
Throughput time	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	5.56	8.41	3.95	…	8.21	9.68	**7.47** (± 0.22)
Replication 2	3.18	6.03	8.67		7.25	8.44	**7.48** (± 0.29)
Replication 3	4.76	5.62	5.78	…	10.36	11.98	**7.63** (± 0.30)
…	…	…	…	…	…	…	…
Replication R	6.00	5.54	3.73	…	5.26	8.15	**7.66** (± 0.24)





Tl						mer n	Average
R						68	**7.47** (± 0.22)
R						44	**7.48** (± 0.29)
R						98	**7.63** (± 0.30)
…	…	…	…	…	…	…	…
Replication R	6.00	5.54	3.73	…	5.26	8.15	**7.66** (± 0.24)

The Petrol Station – Pilot runs



10 runs of 15 days

Pilot run	Avg. throughput time	# Gone	# Total	Percentage
1	7.47 (± 0.22)	454	5369	8,46%
2	7.48 (± 0.29)	519	5422	9,57%
3	7.63 (± 0.30)	526	5425	9,70%
4	7.37 (± 0.40)	605	5562	10,88%
5	7.43 (± 0.19)	523	5346	9,78%
6	7.63 (± 0.25)	480	5436	8,83%
7	7.59 (± 0.25)	588	5504	10,68%
8	7.64 (± 0.25)	533	5403	9,86%
9	7.54 (± 0.28)	517	5366	9,63%
10	7.66 (± 0.24)	538	5403	9,96%

Pilot runs

A pilot run is a simulation run of reasonable length (without having determined the simulation parameters yet)

Suitable for verification and validation, but not for the actual simulation study

- Need to do a few (e.g. 5-10)

For the actual simulation study you have to decide on the simulation parameters (will be discussed in STEP 6)

- Run length
- Number of replications
- When to start and end measurement in a replication
- Whether replications are independent

Business Process Simulation

5b) Simulation experiments and output analysis

Simulation Methodology (7 steps)

STEP 1: Project definition

- **Parameters for experiments: replication length, number of replications, warm-up period, etc.**
- **Output analysis: statistically reliable data, valid conclusions**

STEP 2: Design the simulation study

STEP 3: Conceptual model

STEP 4: Executable model and verification

STEP 5: Validation

STEP 6: Experiments and output analysis

STEP 7: Conclusion

Why do replications differ?

	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	0.00	0.00	0.50	…	3.81	2.56	**7.47** (± 0.22)
Replication 2	0.00	0.03	1.67		7.25	8.44	**7.48** (± 0.29)
Replication 3	0.00	2.62	5.78	…	10.36	11.98	**7.63** (± 0.30)
…	…	…	…	…	…	…	…
Replication R	0.00	1.54	0.73	…	14.26	12.15	**7.66** (± 0.24)

“Random in – Random out”

Environmental variables, e.g.:
- Interarrival times: Expo(4)
- Processing times: U(1,3)

Decision chances

…

Real object system

Output variables, e.g.:
- Average throughput time
- Average waiting time
- …

Random input:
- Random numbers
- Random variates

⇨

Simulation model

⇨

Random output:
- Performance measures

Simulation

Most serious disadvantage of simulation:

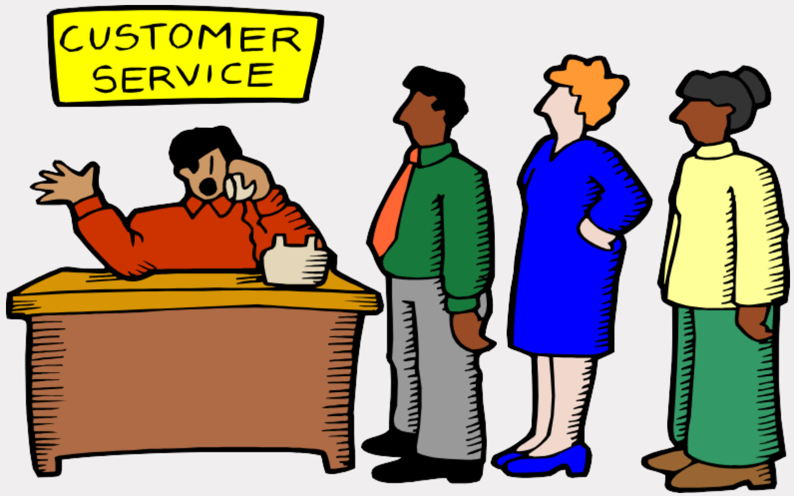
- With a stochastic simulation, you don't get exact "answers" from a run
- Two different runs of same model ⇒ different numerical results

Failure to deal with randomness in simulation output can lead to serious errors, misinterpretation, bad decisions !!!

Statistical analysis needed (for input and output)!

“Random in – Random out”

Input analysis:
Finding fitting distributions for: Inter arrival times, Service times, Chances, etc.



Real object system

Output analysis:
“ sample” of population, needs statistical analysis (use of appropriate methods)

Random input:
- Random numbers
- Random variates

Simulation model

Random output:
- Performance measures

Input analysis

Finding Fitting Distributions:

- From empirical data
  - Histograms
  - Goodness-of-Fit tests
- From no data

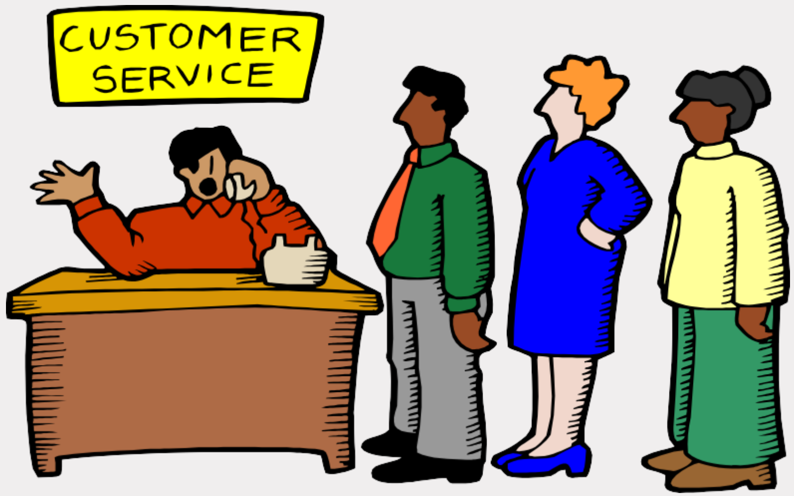
Sensitivity analysis may be useful too

More about input analysis?

- Read chapter 9 of the book

“Random in – Random out”

Input analysis:
Finding fitting distributions for: Inter arrival times, Service times, Chances, etc.



Real object system

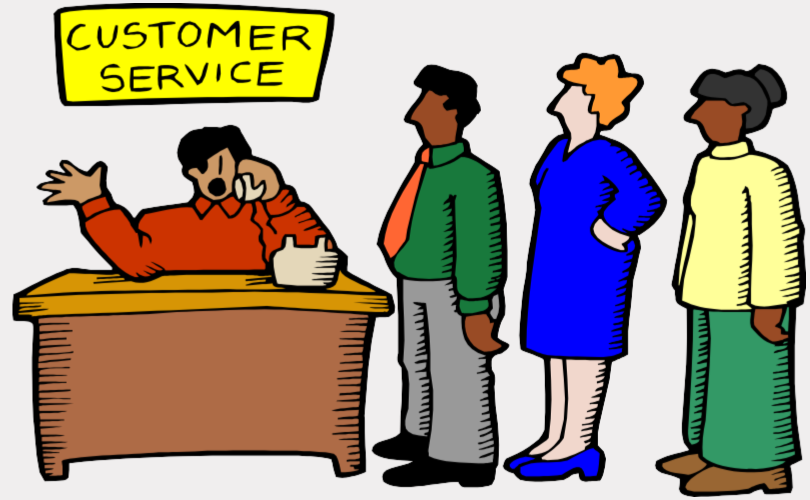
Output analysis:
“ sample” of population, needs statistical analysis (use of appropriate methods)

Random input:
- Random numbers
- Random variates

Simulation model

Random output:
- Performance measures

Simulation output analysis



	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	0.00	0.00	0.50	…	3.81	2.56	**7.47** (± 0.22)
Replication 2	0.00	0.03	1.67		7.25	8.44	**7.48** (± 0.29)
Replication 3	0.00	2.62	5.78	…	10.36	11.98	**7.63** (± 0.30)
…	…	…	…	…	…	…	…
Replication R	0.00	1.54	0.73	…	14.26	12.15	**7.66** (± 0.24)

Output Analysis

“Sample” of population

=> performance measurements in one replication are dependent on each other (autocorrelation)

=> use several replications

Output needs statistical analysis

- Use of appropriate methods, e.g.
  - Point estimates (mean and standard deviation)
  - Interval estimates (confidence intervals)
- Comparison of alternatives

Simulation output - Statistical Analysis

Statistical inference: set of methods to assist a decision maker to draw conclusions about a whole population from a specific sample

1. Estimation involves establishing a degree of accuracy associated with a **point estimate** (what is the difference between μ the theoretical true mean of a distribution, and a point estimate of a distribution’s true mean)

2. **Hypothesis testing** is used in decision making (trying to make a correct decision regarding a pre-stated supposition. Statistical analysis provides information for accepting, or rejecting a hypothesis, see the Chi-square test)

Statistical analysis - Point Estimates

Quantities that are characteristics of a population/distribution, such as μ (mean) or σ^2 (variance), are called parameters

- Generally these population parameters are unknown

The best we can do is to estimate these parameters from a sample.

Since we are estimating a parameter by a single number, this process is referred to as point estimation

Simulation output analysis

	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	0.00	0.00	0.50	…	3.81	2.56	**7.47** (±0.22)
Replication 2	0.00	0.03	1.67		7.25	8.44	**7.48** (±0.29)
Replication 3	0.00	2.62	5.78	…	10.36	11.98	**7.63** (±0.30)
…	…	…	…	…	…	…	…
Replication R	0.00	1.54	0.73	…	14.26	12.15	**7.66** (±0.24)

Simulation output analysis

$$\bar{Y} = \frac{\sum_{i=1}^{R} \bar{Y}_i}{R} = \quad 7.50 \quad (\pm\ 0.15)$$

	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	0.00	0.00	0.50	…	3.81	2.56	**7.47** (±0.22)
Replication 2	0.00	0.03	1.67		7.25	8.44	**7.48** (±0.29)
Replication 3	0.00	2.62	5.78	…	10.36	11.98	**7.63** (±0.30)
…	…	…	…	…	…	…	…
Replication R	0.00	1.54	0.73	…	14.26	12.15	**7.66** (±0.24)

Simulation output analysis

$$\bar{Y} = \frac{\sum_{i=1}^{R} \bar{Y}_i}{R}$$

	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	Y_{11}	Y_{12}	Y_{13}	…	$Y_{1(n-1)}$	Y_{1n}	$\bar{Y}_1$
Replication 2	Y_{21}	Y_{22}	Y_{23}		$Y_{2(n-1)}$	Y_{2n}	$\bar{Y}_2$
Replication 3	Y_{31}	Y_{32}	Y_{33}	…	$Y_{3(n-1)}$	Y_{3n}	$\bar{Y}_3$
…	…	…	…	…	…	…	…
Replication R	Y_{R1}	Y_{R2}	Y_{R3}	…	$Y_{R(n-1)}$	Y_{Rn}	$\bar{Y}_R$

Estimation population parameters

Population parameter	Sample estimate
μ	$\overline{X} = \dfrac{\sum_{i=1}^{n} X_i}{n}$
σ^2	$S^2(n) = \dfrac{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2}{n-1}$

Confidence intervals

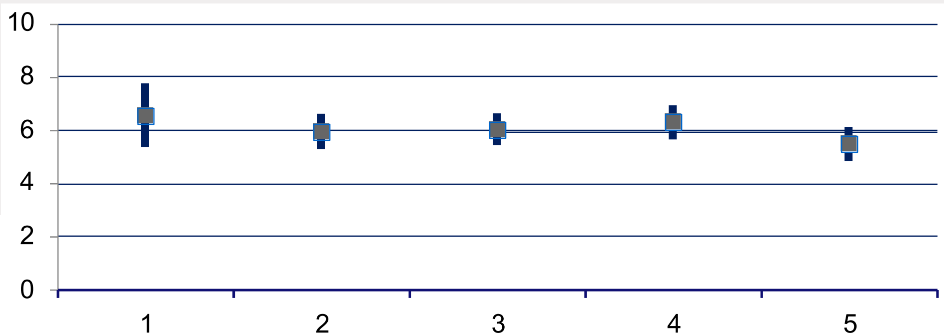
A confidence interval is an interval estimate for a parameter that specifies a level of confidence

$(a < \mu < b)$

Provides a way of quantifying imprecision

The interval contains, or "covers" the target parameter with a pre-specified (high) probability called the confidence level

As the probability becomes higher, the length of the interval becomes larger
It may be better to be 95% confident that $10 < \mu < 15$, than 99% confident that $6 < \mu < 19$



Confidence Intervals

A (1-α)% confidence interval for the mean is a range of values running from a lower bound (LB) to an upper bound (UB) for which we can be (1-a)% confident that the true population mean falls in the interval

- e.g., If we want to have 95% confidence that the mean is a range of values running from a lower bound LB to an upper bound UB, we use α = 0.05

Confidence Intervals for the mean when the population standard deviation are estimated from the data:

$$LB = \overline{X} - \left(t_{df,1-\alpha/2}\right)\frac{s}{\sqrt{n}} \qquad UB = \overline{X} + \left(t_{df,1-\alpha/2}\right)\frac{s}{\sqrt{n}}$$

where s = the standard deviation:

$$\sqrt{\frac{\sum_{i=1}^{n}[X_i - \overline{X}]^2}{n-1}}$$

TU/e

Hypothesis testing

If confidence intervals overlap you may want to use hypothesis testing to test if the means are really different

- H0: there is no difference in average waiting time in the old and new situation
- H1: average waiting time in the old situation is higher than in the new situation

The Central Limit Theorem

- As we increase the number of samples (N > 30), the distribution of the means will be approximately normally distributed

- Then you can use statistical tests for normally distributed data

Consequences for simulation

- Each single run of a stochastic simulation model should be considered as a single sample
  - Each independent model replication, where replications are performed using different random number streams, produces another sample point.
  - Measurements within a single run are dependent !!!

=> NEED TO MAKE SURE REPLICATIONS ARE INDEPENDENT

- If confidence intervals are too big we can not draw any conclusion

=> NEED TO DEFINE HOW MANY REPLICATIONS TO RUN

Simulation Output Analysis

Three statistical assumptions:

Observations are independent

- No correlation exists between consecutive observations

Observations are identically distributed

- Follow the same distribution throughout the entire duration

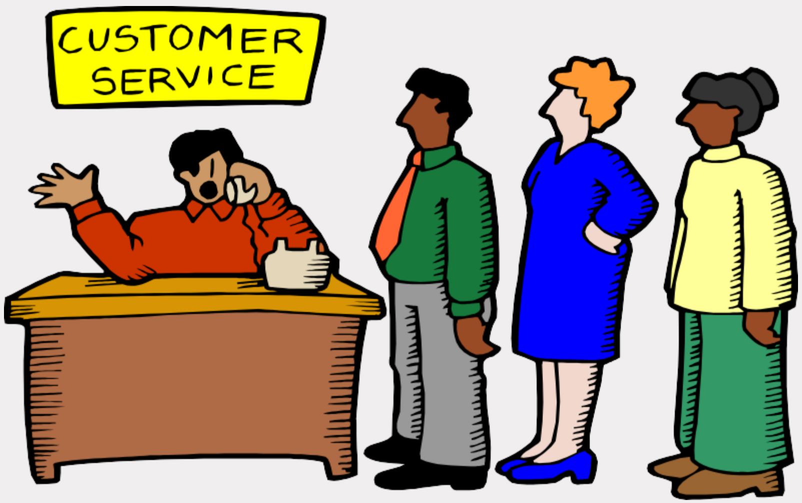
Observations are normally distributed

- Statistical analysis techniques used

=> Correct set-up of simulation study is very important!!!

Simulation experiment set-up

$$\overline{Y} = \frac{\sum_{i=1}^{R} \overline{Y}_i}{R}$$



	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	Y_{11}	Y_{12}	Y_{13}	…	$Y_{1(n-1)}$	Y_{1n}	$\mathbf{Y_1}$
Replication 2	Y_{21}	Y_{22}	Y_{23}		$Y_{2(n-1)}$	Y_{2n}	$\mathbf{Y_2}$
Replication 3	Y_{31}	Y_{32}	Y_{33}	…	$Y_{3(n-1)}$	Y_{3n}	$\mathbf{Y_3}$
…	…	…	…	…	…	…	**…**
Replication R	Y_{R1}	Y_{R2}	Y_{R3}	…	$Y_{R(n-1)}$	Y_{Rn}	$\mathbf{Y_R}$

Simulation experiment set-up

How to set up the experiment:

- Run length of a replication? (n / simulation time)
- Number of replications? (R)
- Independent replications?
- Start and end of measurement (warm up / cool down period)?

	Customer 1	Customer 2	Customer 3	…	Customer n-1	Customer n	Average
Replication 1	Y_{11}	Y_{12}	Y_{13}	…	$Y_{1(n-1)}$	Y_{1n}	$\mathbf{Y_1}$
Replication 2	Y_{21}	Y_{22}	Y_{23}		$Y_{2(n-1)}$	Y_{2n}	$\mathbf{Y_2}$
Replication 3	Y_{31}	Y_{32}	Y_{33}	…	$Y_{3(n-1)}$	Y_{3n}	$\mathbf{Y_3}$
…	…	…	…	…	…	…	**…**
Replication R	Y_{R1}	Y_{R2}	Y_{R3}	…	$Y_{R(n-1)}$	Y_{Rn}	$\mathbf{Y_R}$

Simulation experiment set-up

How to set up the experiment:

- Run length of a replication?
- Number of replications?
- Independent replications?
- Start and end of measurement (warm up / cool down period)?

First decide on:

- Terminating / steady state simulation?

Simulation Experiments

- Terminating simulation

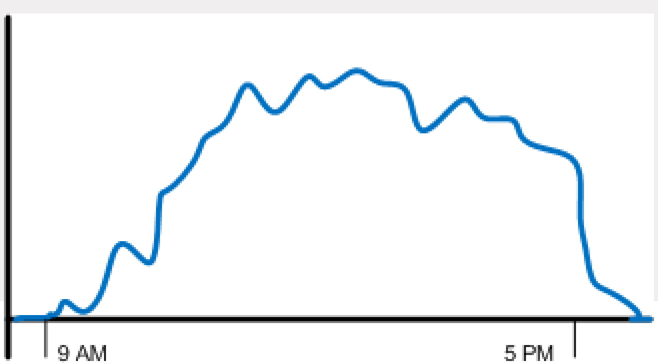
Only interested in the behavior of the system in a particular period

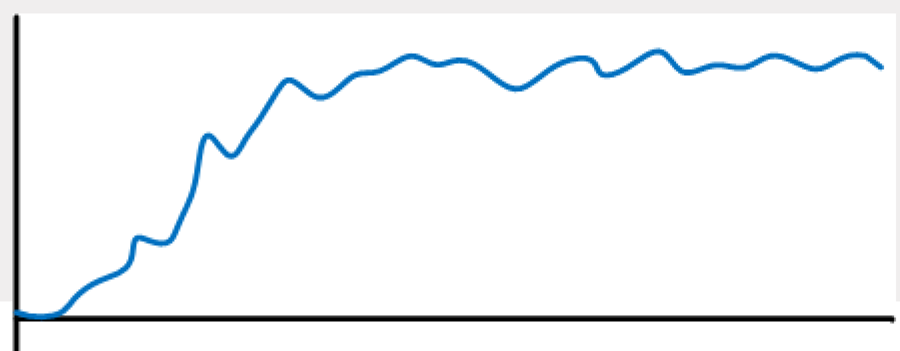
e.g. roadway system during rush hour, opening hours of a shop

- Nonterminating simulation

Interested in the steady-state behavior of a system

e.g. 24/7 production line





Which one is appropriate?

Depends on:
- goals of the study and
- nature of the system
- Statistical analysis for terminating simulations is a lot easier (however modeling the start and end conditions can be a lot more difficult)

Is steady-state relevant at all?
- 24 h/day, “lights-off” manufacturing, or office hours?
- Is system state conserved between days?

Some examples

Physical model	*Terminating estimand*	*Steady-state estimand*
Single-server queue	Expected average delay in queue of first 25 customers given empty-and-idle initial conditions	Long-run expected delay in queue of a customer
Manufacturing system	Expected daily production given some number of workpieces in process initially	Expected long-run daily production

TU/e

Simulation experiment set-up

How to set up the experiment:

- Run length of a replication?
- Number of replications?
- Independent replications?
- Start and end of measurement (warm up / cool down period)?

First decide on:

- Terminating / steady state simulation?

Business Process Simulation

Random number generators

"Random in – Random out"

Environmental variables, e.g.:
- Interarrival times: Expo(4)
- Processing times: U(1,3)

Decision chances

…

Real object system

Output variables, e.g.:
- Average throughput time
- Average waiting time
- …

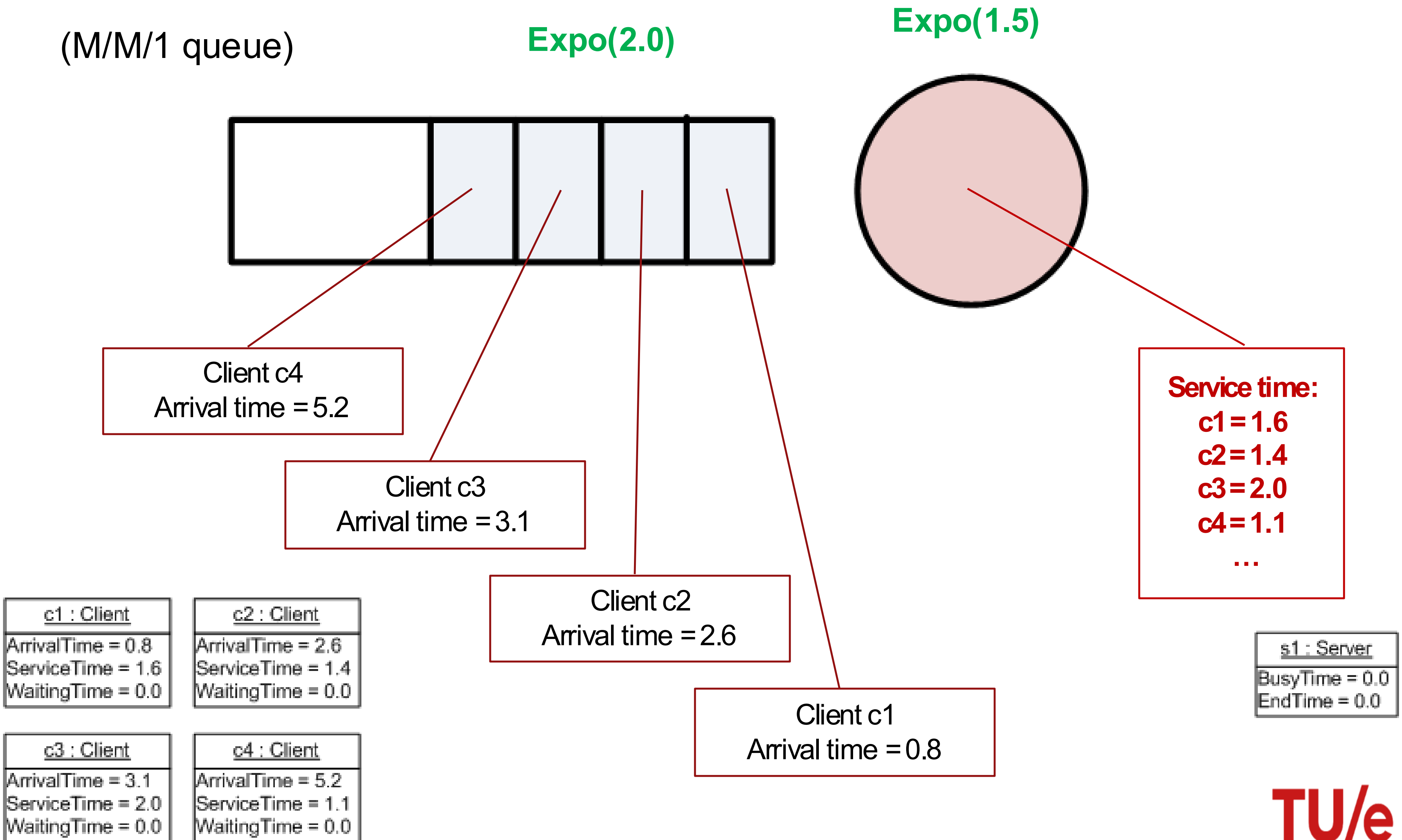
Random input:
- Random numbers
- Random variates

⇨ **Simulation model** ⇨

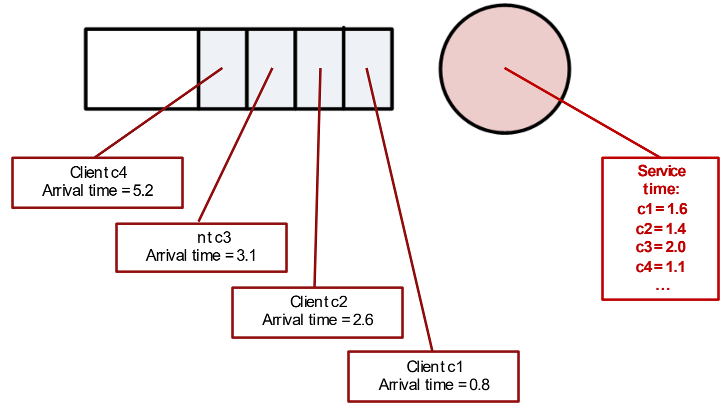
Random output:
- Performance measures

How does simulation work?

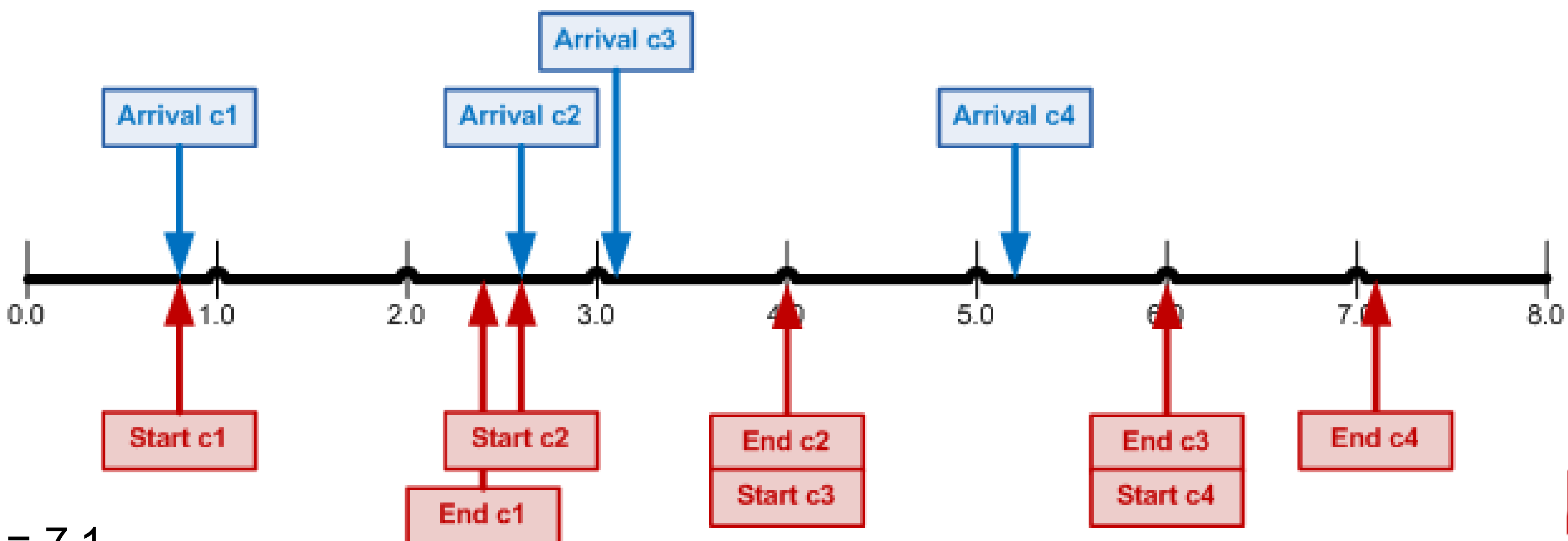
(M/M/1 queue)



Time line



- **Next arrival event:**
  - -
- **Next server event:**
  - -



time = 7.1

Random Number Generators

1) All stochastic simulations “generate” IID U(0,1) “random numbers” somehow

2) These generated IID U(0,1) numbers are then transformed into a number from the given distribution (Expo(2.0), U[2, 8], etc.)

PSEUDO Random Number Generation

PROPERTIES OF TRUE RANDOM NUMBERS

A sequence of random numbers r1, r2, ….must have two statistical properties:

- uniformity and
- independence.

each random number is an independent sample drawn from a continuous uniform distribution between 0 and 1.

There are no true random number generators!!

- If the method is known, the set of random numbers can be replicated…The numbers are not truly random!

GOAL: Generate a sequence of numbers between 0 and 1 which simulates or imitates the ideal properties of uniform distribution and independence as closely as possible.

TU/e

PSEUDO Random Number Generation

Criteria for good pseudo random number generation methods:

- Fast routine
- Portability
- A sufficiently long cycle
- Random numbers replicable
- Closely approximate the ideal statistical properties of uniformity and independence

It is HARD to invent techniques!

The mostly used one is the

LINEAR CONGRUENTIAL METHOD (by Lehmer 1951)

THE LINEAR CONGRUENTIAL METHOD

Produces a sequence of integers X1, X2,…between 0 and m-1 according to:

$$X_{i+1} = (aX_i + c) \bmod (m), i = 0,1,2,...$$

$$\text{and } R_i = X_i / m \ (i=1,2,\ldots)$$

Where:

X_0: The initial value is called the *seed*.

a: constant multiplier,

c: increment,

m: modulus.

a, c, and m are non negative integers and must satisfy m>0, m>a, m>c and Xo <m

TU/e

THE LINEAR CONGRUENTIAL METHOD

$$X_{i+1} = (a \cdot X_i + c)\, mod\, (m)$$

$$R_i = \frac{X_i}{m}$$

Example:

Generate a sequence of three random numbers

with X_0= 1, a= 6, c= 1 and m= 25.

$X_1 = (6*1+1)\mathrm{mod}(25)$	$X_1 = 7$	$R_1 = 7/25 = 0.28$
$X_2 = (6*7+1)\mathrm{mod}(25)$	$X_2 = 18$	$R_2 = 18/25 = 0.72$
$X_3 = (6*18+1)\mathrm{mod}(25)$	$X_3 = 9$	$R_3 = 9/25 = 0.36$

$R_4 = ?$

THE LINEAR CONGRUENTIAL METHOD

$$X_{i+1} = (a \cdot X_i + c)\, mod\, (m)$$

$$R_i = \frac{X_i}{m}$$

__Example:__

Generate a sequence of three random numbers

with X_0= 1, a= 6, c= 1 and m= 25.

$X_1 = (6*1+1) \text{mod} (25)$	$X_1 = 7$	$R_1 = 7/25 = 0.28$
$X_2 = (6*7+1) \text{mod} (25)$	$X_2 = 18$	$R_2 = 18/25 = 0.72$
$X_3 = (6*18+1) \text{mod} (25)$	$X_3 = 9$	$R_3 = 9/25 = 0.36$

$X_4 = (6*9 + 1)\ mod\ (25)$ $X_4 = 5$ $R_4 = 5/25 = 0.20$

Drawing numbers from a desired distribution

- First a IID U(0,1) number is generated
- Then it's transformed into desired distribution, e.g.:

IID U(0,1)

- R1 = 0.28
- R2 = 0.72
- R3 = 0.36
- R4 = 0.20

…

Uniform[0,15]:

- R1 = round (0.28 * 15) = 4
- R2 = round (0.72 * 15) = 11
- R3 = round (0.36 * 15) = 5
- R4 = round (0.20 * 15) = 3

…