

Confidence Intervals and Replications in AnyLogic

The Medical Practice – A Business Perspective

Computing Confidence Intervals

The method of *independent* replications:

- Run the simulation n times
- Use independent sets of random numbers
- Make the observations $Y_r, r = 1, \dots, n$
- Compute an estimate $\hat{\theta}$ for the measure of interest
- Compute an estimate S^2 for the variance of this estimator

Confidence Interval Computation

Estimator for mean value

$$\hat{\theta} = \frac{1}{n} \sum_{r=1}^n Y_r$$

Estimator for standard deviation

$$\hat{\sigma}(\hat{\theta}) \approx \sqrt{\frac{S^2}{n}} = \sqrt{\frac{\sum_{r=1}^n (Y_r - \hat{\theta})^2}{n(n-1)}}$$

Symbols:

- $n \dots$ number of replications
- $Y_r \dots$ value of the measured variable at the r th replication

Confidence Interval Computation

Confidence Interval

$$\hat{\theta} - \hat{\sigma}(\hat{\theta}) \cdot t_{\alpha/2, f} \leq \theta \leq \hat{\theta} + \hat{\sigma}(\hat{\theta}) \cdot t_{\alpha/2, f}$$

Symbols:

- $t_{\alpha/2, f}$... value of the inverse Student's t-CDF (“*TINV*”)
 - $f = n - 1$... degrees of freedom
 - α ... level of significance

Confidence Interval Interpretation

A useful result could look like this:

$$0.1 \leq \theta \leq 0.6 \quad \text{with } \alpha = 0.01$$

And means: “The true value of θ lies inside the interval $[0.1, 0.6]$ with certainty **0.99** ($= 1 - \alpha$)”

Pitfall:

- To obtain the necessary value of the t-distribution, use $\alpha/2$, not α !

Experiments in AnyLogic

Experiment Types in AnyLogic PLE

Simulation

- Simulates the model exactly once (So far, we only used this)

Parameter Variation

- Multiple replications for multiple versions of a model

Optimization

- Automatically optimize model parameters with respect to a user-defined objective function

Experiments in AnyLogic

To compute confidence intervals, we need to:


- Run the model multiple times
- Analyse the results statistically

Both tasks are tedious to do manually

AnyLogic experiments enable us to write custom Java code to be executed:







- After each replication has finished (e.g. to store the results)
- After all replications of a parameter set (iteration) have finished (e.g. to automatically compute confidence intervals)

Parameter Variation Experiment

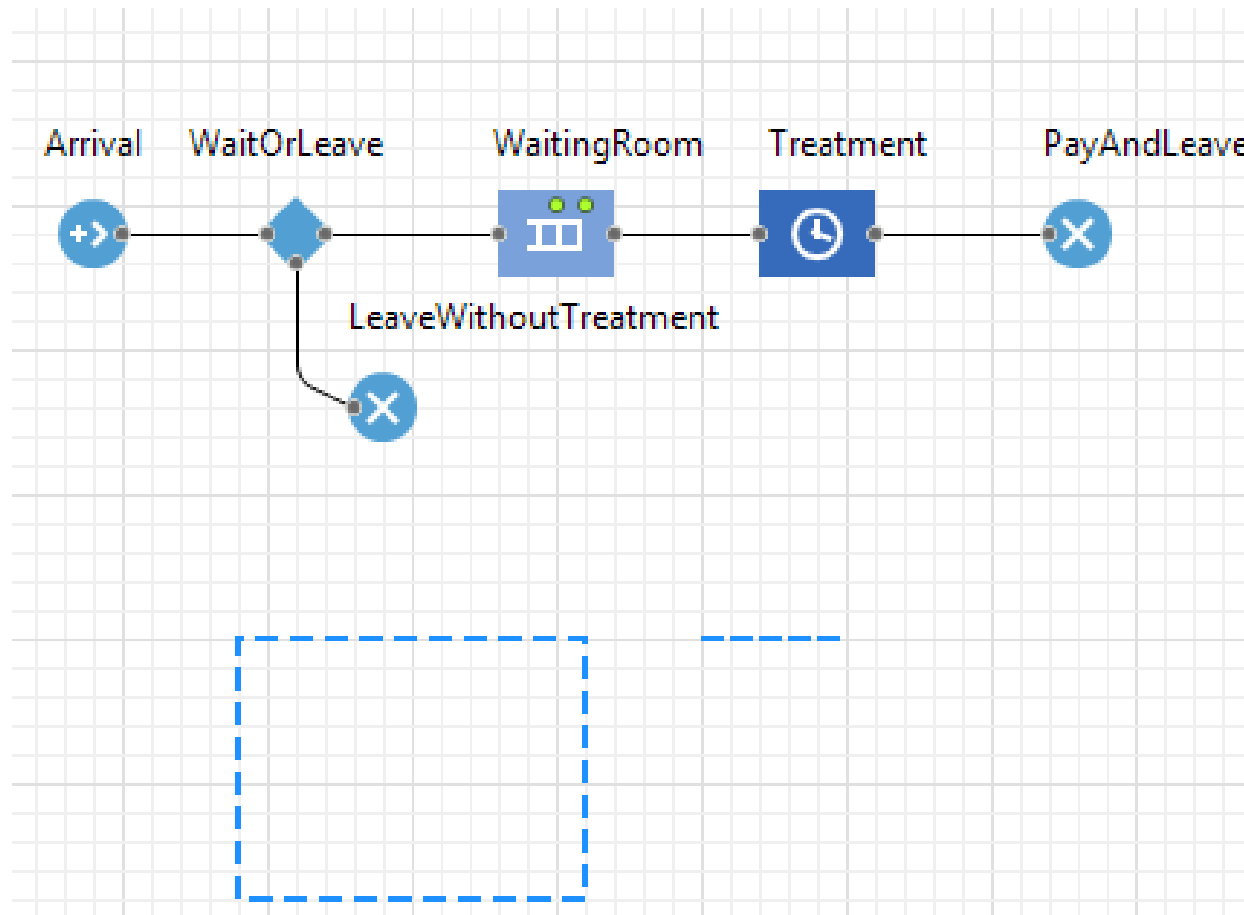
Properties 

ParametersVariation1 - Parameter Variation Experiment

Java actions

Initial experiment setup:		Executed once, before everything else
Before each experiment run:		Executed once for each parameter set
Before simulation run:		Executed once for each simulation
After simulation run:		Executed once for each simulation
After iteration:		Executed once for each parameter set
After experiment:		Executed once, after everything is done

Example: The Waiting Room Model



Example: The Waiting Room Model

We again use the waiting room model with

- Inter-Arrival times: exponential(1 / 5.0) minutes
- Treatment duration: normal(2, 4.5) minutes

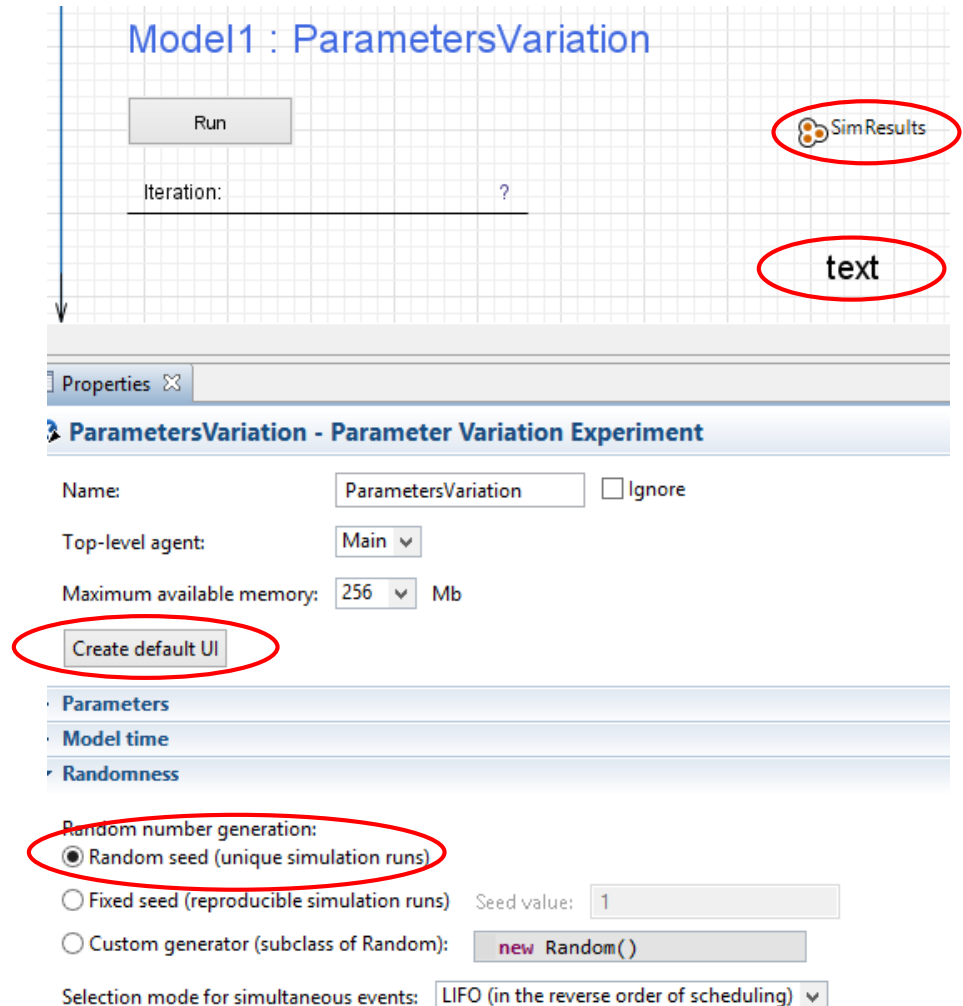
We want to know the waiting queue length at $T=10,000$

This time, we will compute a confidence interval

We will use experiments and custom code to automate the computation

Experiment Setup

- Create a “Parameter Variation Experiment”
- Let AnyLogic “Create default UI”
- Add an (*int*) collection variable to store the replications’ results
- Add a *text* for displaying the results
- Select “unique simulation runs”, otherwise all replications will have the same result!



Model1 : ParametersVariation

Run

Iteration: ?

SimResults

text

Properties

ParametersVariation - Parameter Variation Experiment

Name: ParametersVariation ☐ Ignore

Top-level agent: Main

Maximum available memory: 256 Mb

Create default UI

Parameters

Model time

Randomness

Random number generation:

☒ Random seed (unique simulation runs)

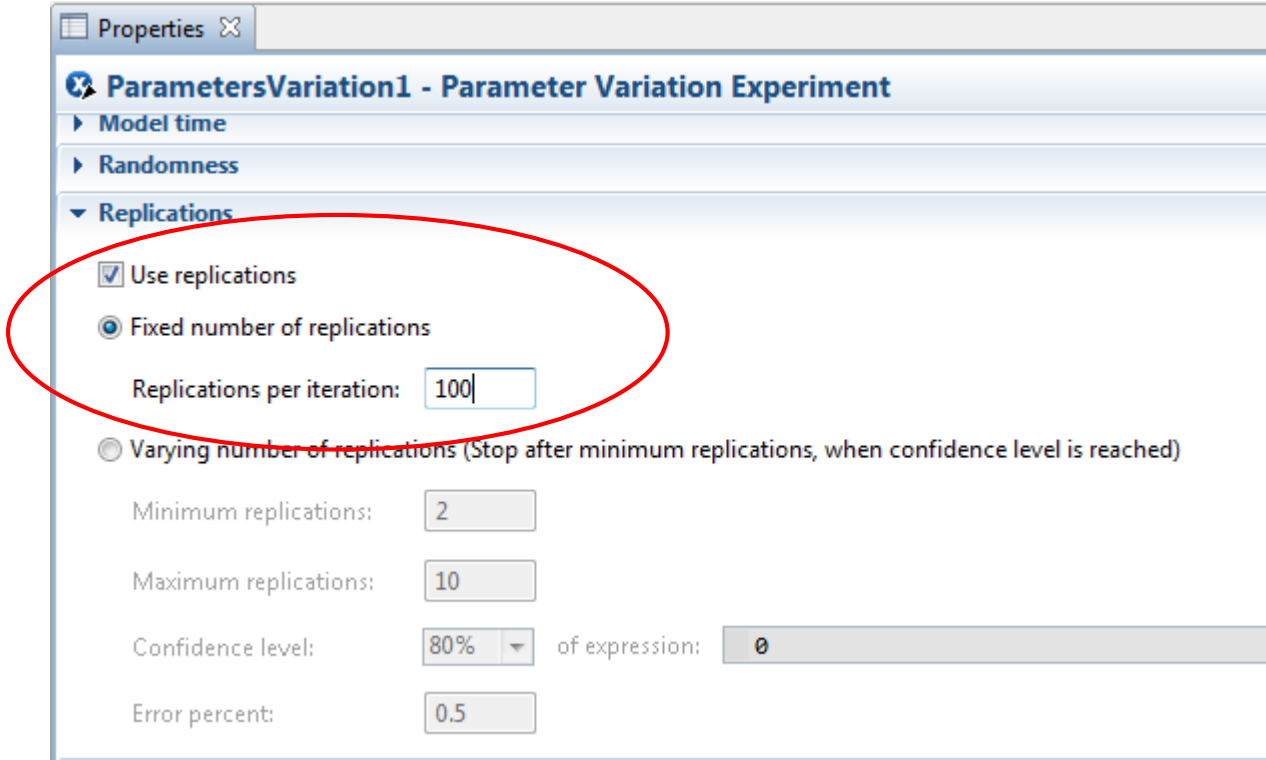
☐ Fixed seed (reproducible simulation runs) Seed value: 1

☐ Custom generator (subclass of Random): new Random()

Selection mode for simultaneous events: LIFO (in the reverse order of scheduling)

Experiment Setup

Set the number of replications to be executed



Properties

ParametersVariation1 - Parameter Variation Experiment

- ▶ Model time
- ▶ Randomness
- ▼ Replications
 - ☒ Use replications
 - ☒ Fixed number of replications
 - Replications per iteration:
 - ☐ Varying number of replications (Stop after minimum replications, when confidence level is reached)
 - Minimum replications:
 - Maximum replications:
 - Confidence level: of expression:
 - Error percent:

The Evaluation Code

„After simulation run“Code:

```
SimResults.add(root.WaitingRoom.size());
```

„After iteration“Code:

```
double sum = 0;
int n = SimResults.size();           //container holding all results
for (int idx = 0; idx < n; idx++) sum += SimResults.get(idx);
    double mean = sum / n;

double sumdiffsq = 0;                //sum of squared differences
for (int idx = 0; idx < n; idx++)
    sumdiffsq += (SimResults.get(idx) - mean) * (SimResults.get(idx) - mean);

double S_squared = sumdiffsq / (n-1);

double tVal = 2.87130765;            //t-value for alpha = 0.005; 99 D.o.F.
double ci_min = mean - tVal * sqrt( S_squared / n);
double ci_max = mean + tVal * sqrt( S_squared / n);

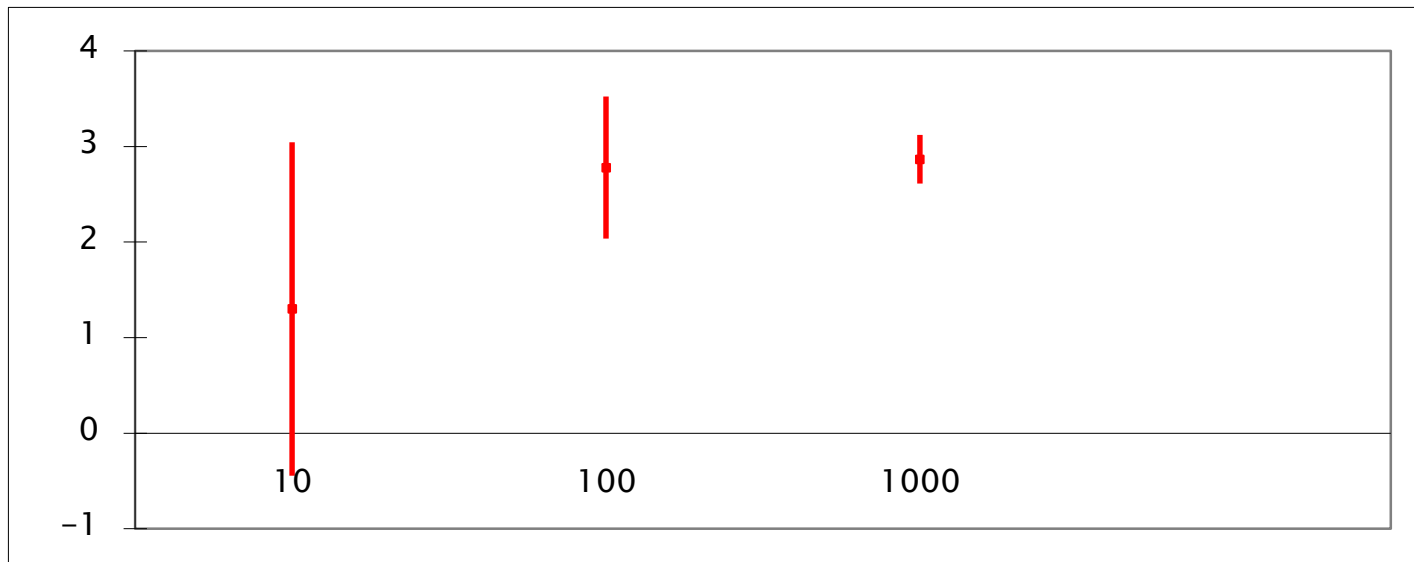
text3.setText( format(ci_min) + " <= theta <= " + format(ci_max) );
```

Results

Queue lengths at time $T=10000$:

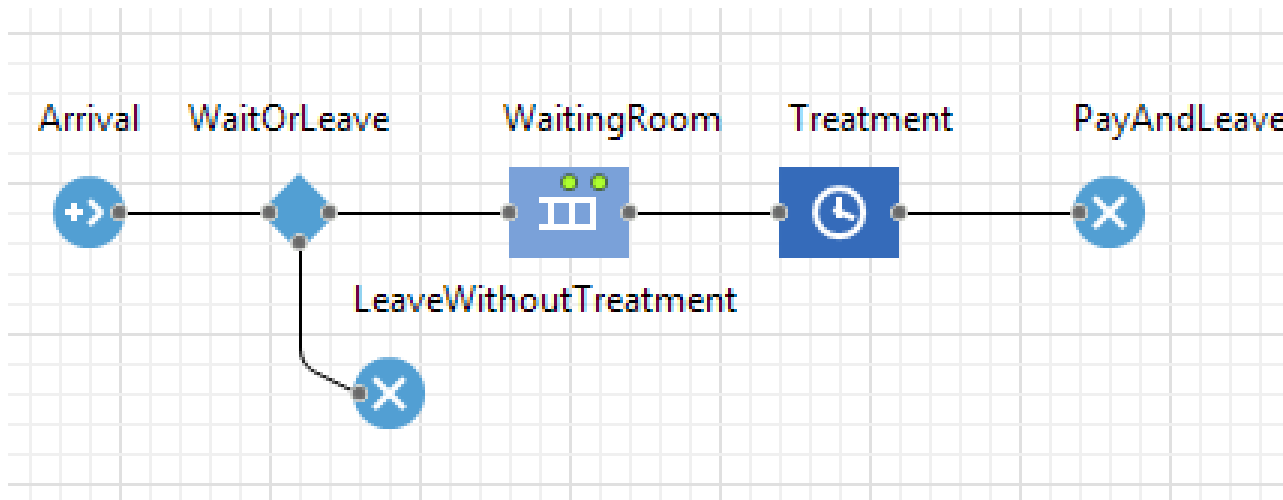
Seed	1	2	3	4	5	6	7	8	9	10
Length	3	0	1	1	0	3	1	0	4	0

Confidence intervals for $n=10, 100, 1000$:



And now from a Business Perspective...

An Actual Parameter Variation Experiment



Imagine the following situation:

- A waiting room costs 2€ per hour and seat in maintenance
- Each successfully treated patient pays 20€
- Patients without a seat will leave the practice at once

The physician of a new medical practice wants to determine an optimal waiting room size (i.e. the one with maximum expected profit).

Experiment Plan

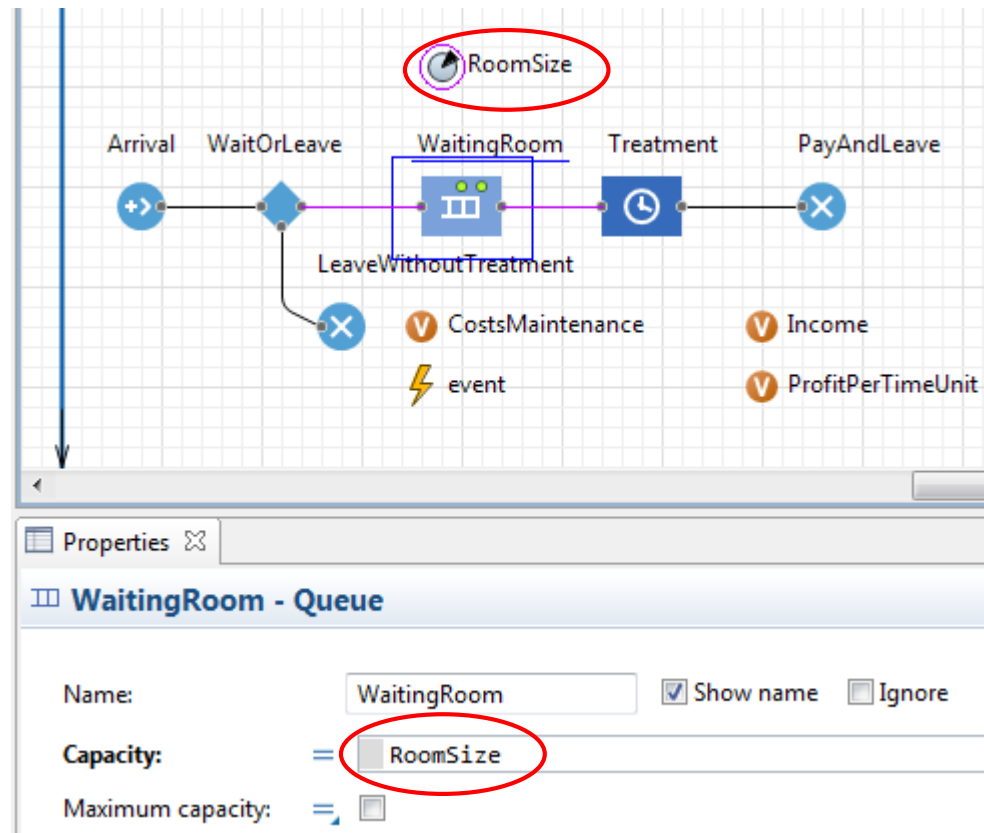
We will

- Vary the parameter *RoomSize* in 20 steps from 1 to 20
→ 20 iterations
- For each iteration, conduct 100 replications and compute a confidence interval for the *ProfitPerTimeUnit*
- Plot each confidence interval and mean against the *RoomSize*

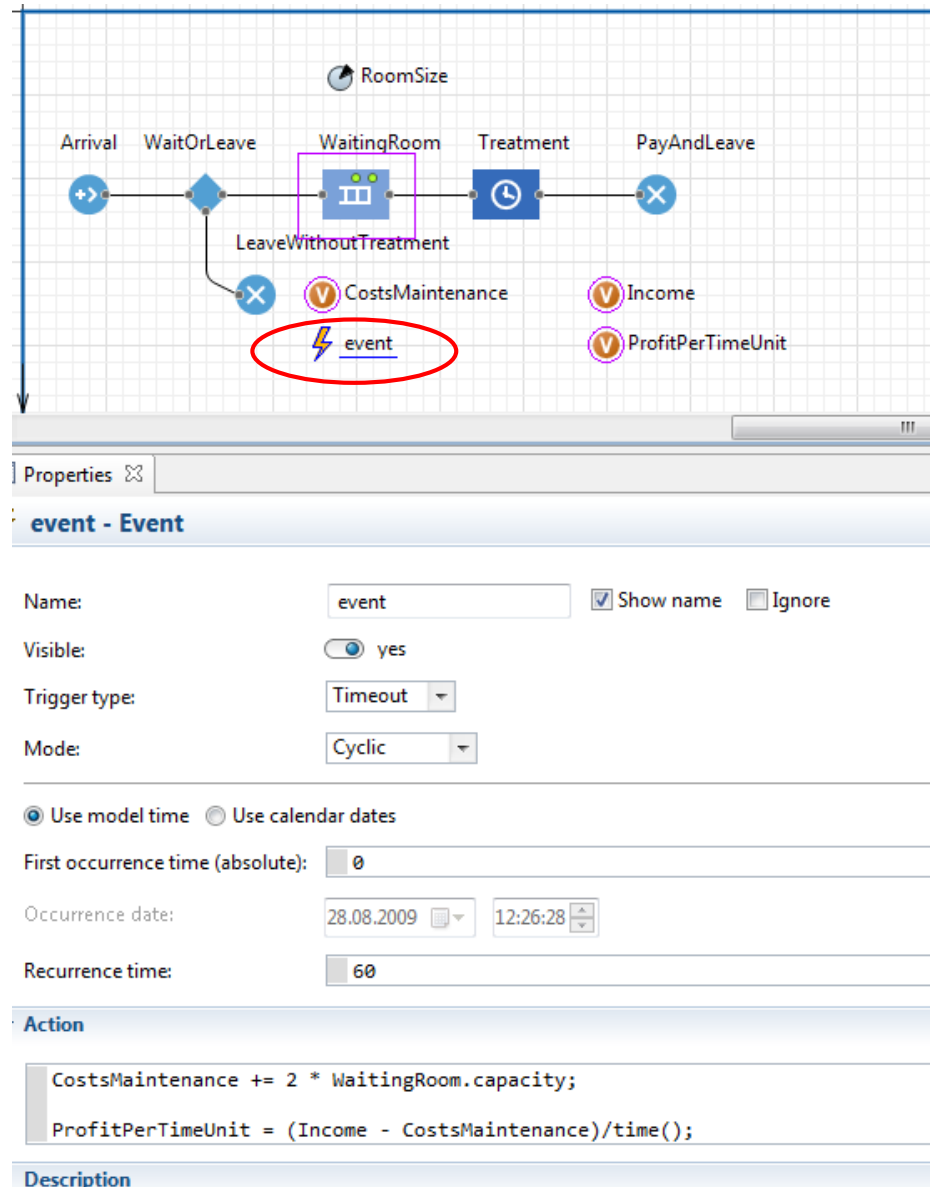
→ We will already have $20 \cdot 100 = 2000$ simulation runs!

Changes to the Model – Parameters

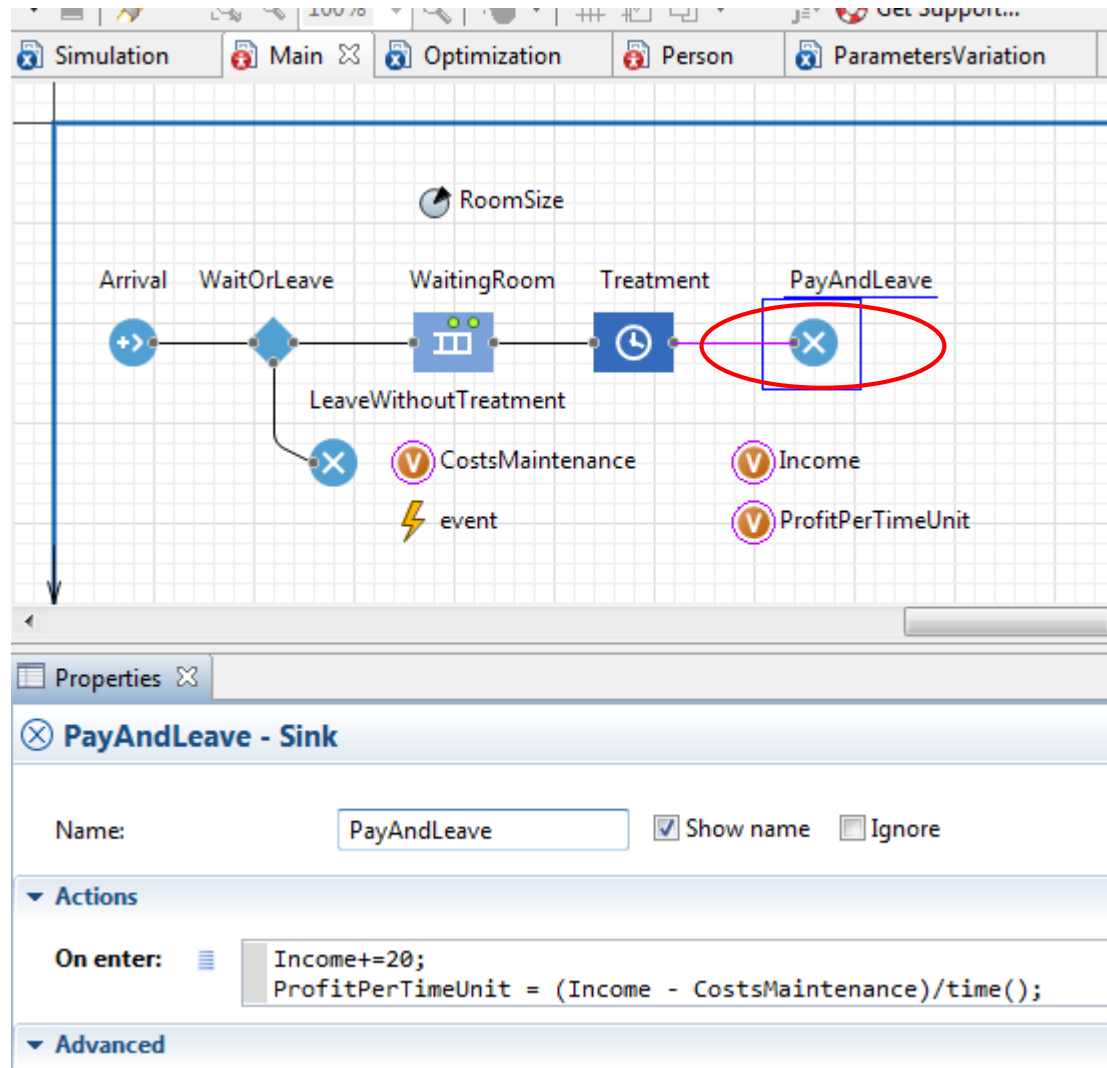
Only parameters can be manipulated (i.e. varied) by the experiment. Inside a model, a parameter can be used the same way as a plain variable.




Changes to the Model – Maintenance Cost Computation




Changes to the Model – Income Computation



Experiment Setup

Properties 

 **ParametersVariation - Parameter Variation Experiment**

Name: ☐ Ignore

Top-level agent: ▼

Maximum available memory: ▼ Mb

▼ **Parameters**

Parameters: ☒ Varied in range ☐ Freeform

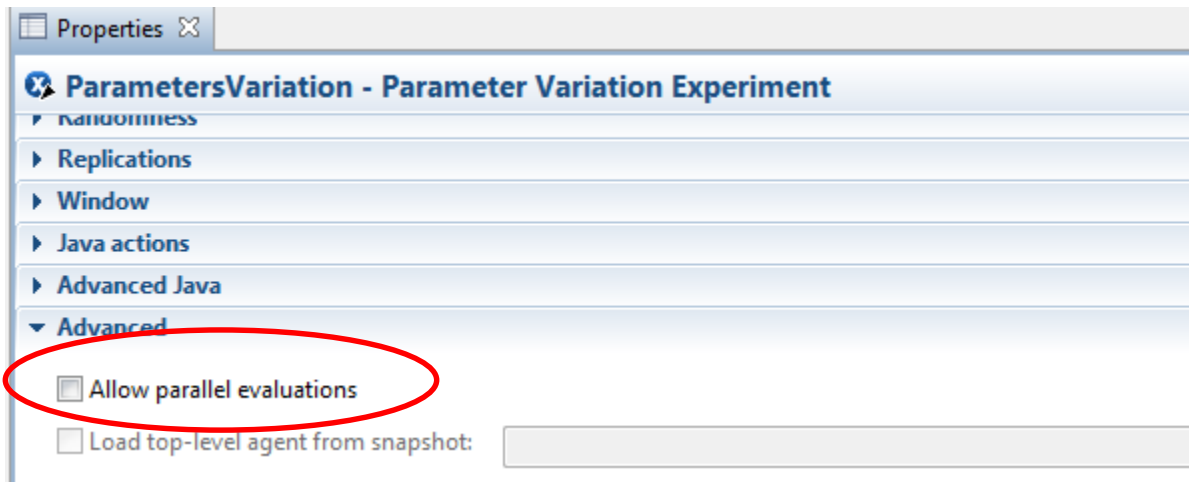
Number of runs:

Parameter	Type	Value		
		Min	Max	Step
RoomSize	Range	1	20	1

Experiment Setup – Pitfall

Using our experiment description we assume that all iterations are executed sequentially !!!

- *Uncheck the “Allow parallel evaluations” checkbox !!*
- Otherwise you will get very strange results 😊



If you want to make use of AnyLogic’s multi-thread ability, be sure to understand the implications of multi-thread programming (race conditions, synchronization, ...)

C.I. Computation – “After Iteration” Code

```
/**
```

```
...
```

```
The usual confidence interval computation code
```

```
*/
```

```
double ci_min = mean - tVal * sqrt( S_squared / n);
```

```
double ci_max = mean + tVal * sqrt( S_squared / n);
```

```
dsCI_Min.add(RoomSize, ci_min);
```

```
dsCI_Max.add(RoomSize, ci_max);
```

```
ds.add( RoomSize, mean);
```

```
simResults.clear();//for the next iteration
```


Diagram Setup

Properties

plot - Plot

Name: ☐ Ignore

▼ Data

☐ Value ☒ Data set

Title:

Data set:

Point style:

Line width: 2 pt

Color:

☐ Value ☒ Data set

Title:

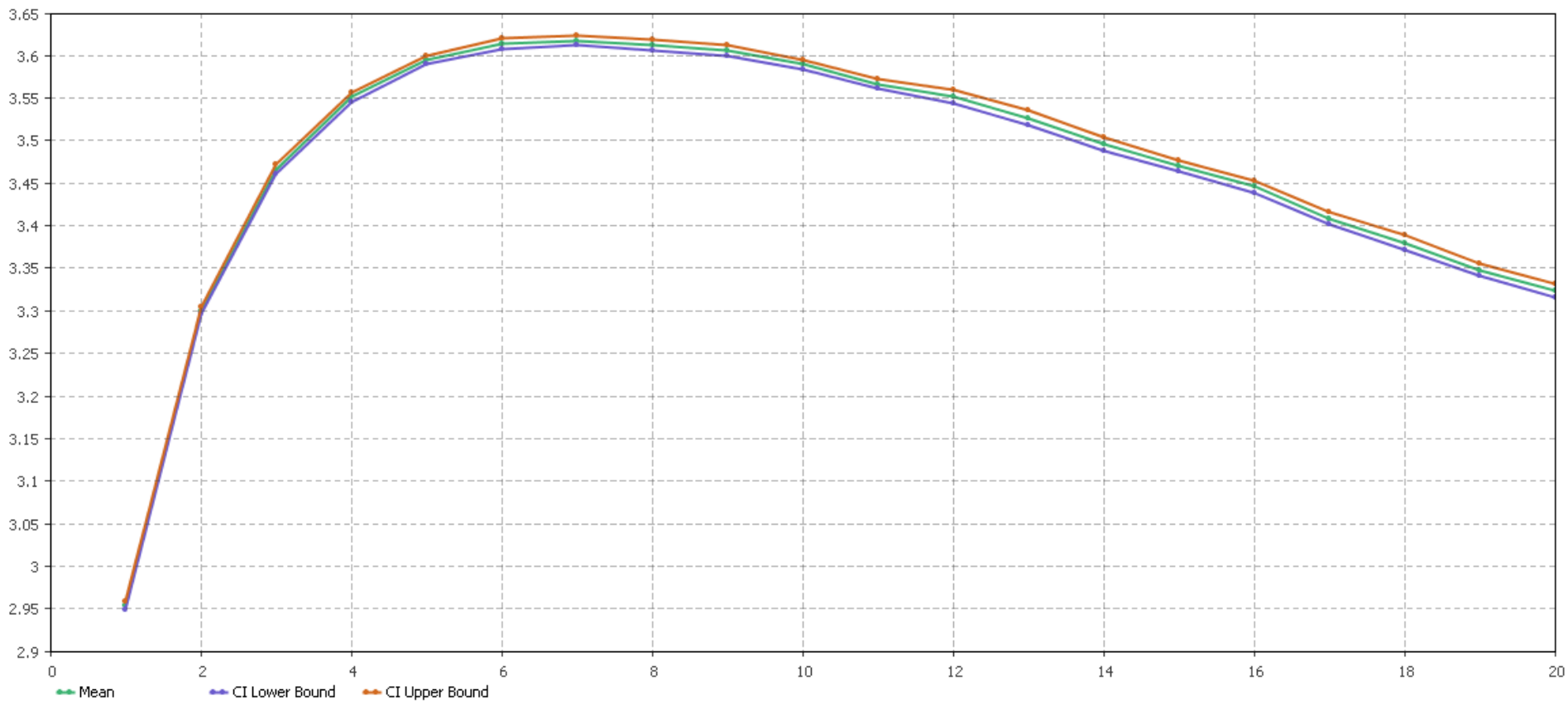
Data set:

Point style:

Line width: 2 pt

Color:

Results



So, what is the optimal waiting room size?

Learning Goals

You are now able to create an experiment to answer the following questions:



What is the expected value of the following measures:



- Shield energy level after two hours
- Number of antimatter particles to hit the shield
- Total time of unemployment of the father
- Amount of money spent on damaged school property

Approach:

- Do at least 100 replications
- Compute the confidence intervals