

# Simulation Product Line

## Research and Development

Fontys University

<b>Date</b>	:	<b>16-09-2024</b>
<b>Version</b>	:	<b>0002</b>
<b>Status</b>	:	<b>Status</b>
<b>Author</b>	:	<b>Joep van Dijk</b>

Version

## *Distribution*

Version	Date	To
0001	28/09/2024	Canvas
0002	11/10/2024	Canvas

*Contents*

Use Case 1.....	6
Assignment 1 .....	6
Assignment 2 .....	6
Standard .....	7
Strategy 1 - Reduced repair time .....	7
Strategy 2 – buffer zones .....	9
Overall results .....	10
Use Case 2.....	11
Assignment 1 .....	11
Assignment 2 .....	12
Strategy 1 .....	13
Strategy 2 .....	15
Strategy 3 .....	18
Strategy 4 .....	20
Strategy 5 .....	22
Results.....	25
Assignment 3 .....	26
Conclusion .....	27
Use Case 3.....	28
Assignment 1 .....	28
Assignment 2 .....	29
layout.....	29
Experimental information .....	30
Results.....	31
Assignment 3 .....	32
Experiment information .....	32
Result .....	32
Assignment 4 .....	33
Experimental information .....	33
Result .....	35
Recommendation .....	35
Conclusion .....	36

Use Case 4.....	37
Assignment 1 .....	37
Analysis.....	39
Design .....	39
Assignment 2 .....	41
Assignment 3 .....	42
Layout .....	42
Assignment 4 .....	43
Assignment 5 .....	43
Research.....	44
Stochastics (random probability distribution).....	44
Discrete distribution.....	44
Continuous distribution.....	44
Empirical Distribution .....	44
Queueing Theory .....	45
M/M/c queueing model.....	45
Experiments .....	45
Performance Measures (pfm).....	45

Figure 1 - strategy 1 layout.....	13
Figure 2 - Experiment info strategy 1 .....	14
Figure 3 - probability distribution strategy 4 .....	15
Figure 4 - strategy 2 layout.....	16
Figure 5 - Label Number of groceries on atom.....	16
Figure 6 - server cycletime [s] .....	16
Figure 7 - Expirement info strategy 2 .....	17
Figure 8 - Strategy 3 layout .....	18
Figure 9 - Source send conditional .....	18
Figure 10 - Experiment info strategy 3 - Exp timing.....	19
Figure 11 - Experiment info strategy 3 – Groceries Count.....	19
Figure 12 - strategy 4 layout .....	20
Figure 13 - Experiment info strategy 4 .....	21
Figure 14 - Strategy 5 layout .....	22
Figure 15 - 4dScript queue entry and exit.....	22
Figure 16 - Trigger settings queue .....	23
Figure 17 - Experiment info strategy 5 .....	24
Figure 18 - Answer results .....	27

Figure 19 - transport system layout .....	29
Figure 20 - Corner Transfer Unit, Entry trigger 4dscript.....	29
Figure 22- Corner Transfer Unit, Exit trigger 4dscript.....	29
Figure 21 - Intersection channel connections .....	29
Figure 23 - Experiment info Uniform(9.5,10.5) .....	30
Figure 24 - Experiment info Uniform(14.5 ,15.5) .....	30
Figure 25 - Experiment Uniform (14.5,15.5).....	30
Figure 26 - Expirment Uniform (9.5,10.5) .....	30
Figure 27 - Expirment info Uniform (10.3,11.3) .....	31
Figure 28 - Expirment Uniform (10.3,11.3).....	31
Figure 29 - Experiment variable production.....	32
Figure 30 - Experiment variable production increased conveyor speed.....	33
Figure 31 - Experiment info variable production increased conveyor speed .....	33
Figure 32 - Experiment variable production transfer unit switch.....	34
Figure 33 - Experiment info variable production transfer unit switch.....	34
Figure 34 - given results use case 3 .....	36
Figure 35 - Production line overview .....	37
Figure 36 - 1st Prototype.....	39
Figure 37 - 2nd Prototype definitive for continuous production .....	40
Figure 38 - Server time settings .....	41
Figure 39 - Continuous production line .....	42
Figure 40 - 3rd Prototype.....	43
 Table 1 - info strategy 1 .....	14
Table 2 - info strategy 2 .....	17
Table 3 - info strategy 3 .....	19
Table 4 - info strategy 4 .....	21
Table 5 - info strategy 5 .....	24
Table 6 - variable production machines .....	32
Table 7 - Variable production increased conveyor speed.....	33
Table 8 - Variable production Transfer unit switch.....	34
Table 9 - Observation dataset.....	38

## Use Case 1

### Assignment 1

Calculated throughput

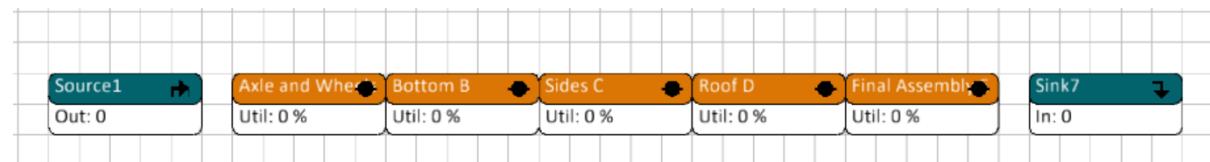
Lowest throughput bottleneck time per hour \* hours per day \* workdays

$$3 * 8 * 5 = 120$$

Loss of production = 10%

$$120 * 0,90 = \underline{108 \text{ trailers per week}}$$

### Assignment 2



## Standard

### Result

Experiment Report

---

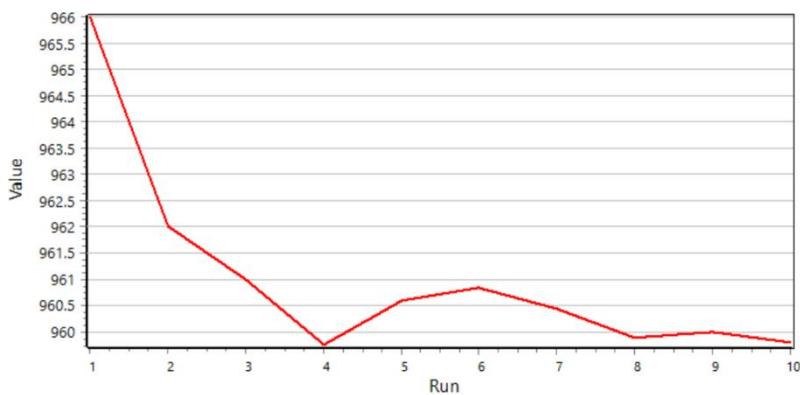
#### Experiment information

Model:	flowshop1.mod
Observation period:	14400000
Warmup period:	144000
Number of replications:	10
Simulation method:	Separate runs
Description:	

#### Sink7

THROUGHPUT	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	959.80	3.36	957.40	962.20	956.00	966.00

#### Cumulative average : Sink7, THROUGHPUT



## Strategy 1 - Reduced repair time

### *Result*

#### Experiment Report

---

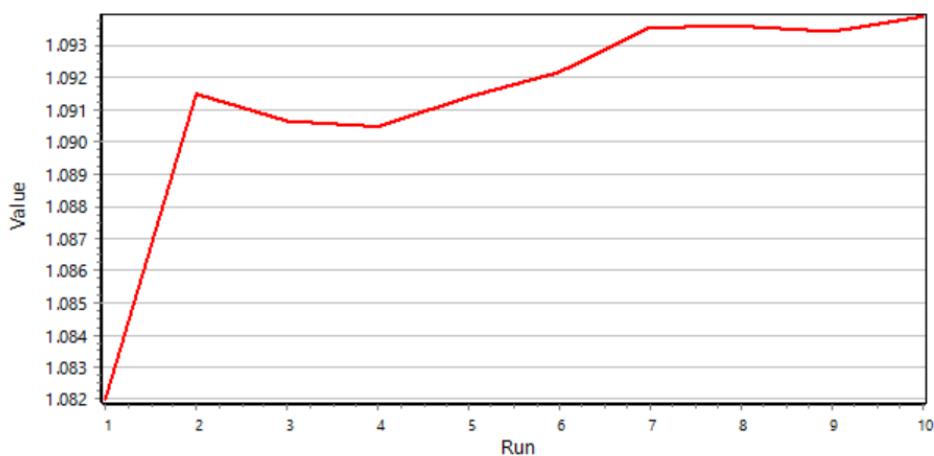
### **Experiment information**

Model: flowshop1.mod  
 Observation period: 14400000  
 Warmup period: 144000  
 Number of replications: 10  
 Simulation method: Separate runs  
 Description:

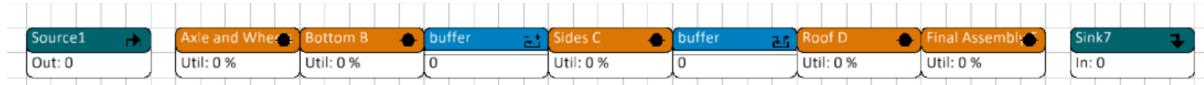
### **Sink7**

<b>THROUGHP</b>	<i>Average</i>	<i>S.Dev.</i>	<i>LB (95%)</i>	<i>UB (95%)</i>	<i>Minimum</i>	<i>Maximum</i>
	1093.90	5.99	1089.62	1098.18	1082.00	1102.00

### **Cumulative average : Sink7, THROUGHPUT**



## Strategy 2 – buffer zones



A 15 min +- 1 min	min 0 - max min 2
B 15 min +- 1 min	min 1 - max min 5
buffer	
C 18 min +- 1 min	min 0 - max min 4
buffer	
D 20 min +- 1 min	bottleneck min 0 - max min 0.5
E 18 min +- 1.5 min	

On the right side you can see the min a max waiting time between each of the server nodes. As you can see the transfer from server B to server C has a minimal transfer time of 1 minute. Because of that it has the highest priority to get a buffer to reduce transfer times. After that we look at the max possible transfer times. The transfer from server C to server D has the second max transfer time. So, this transfer has the second highest priority to get a buffer for transfer time reduction.

### Results

#### Experiment Report

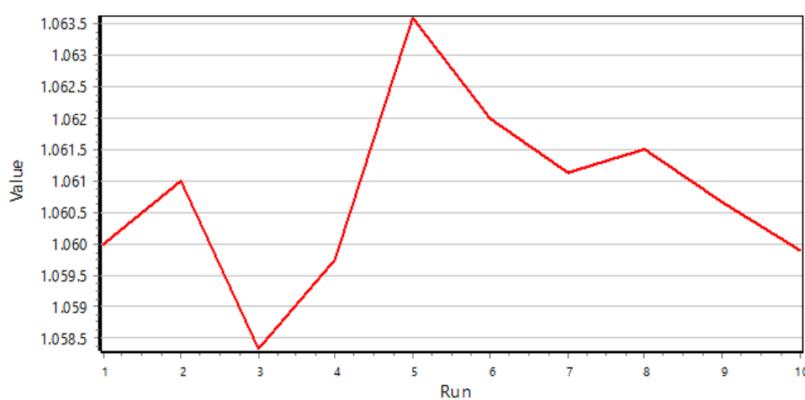
##### Experiment information

Model:	flowshop1.mod
Observation period:	14400000
Warmup period:	144000
Number of replications:	10
Simulation method:	Separate runs
Description:	

##### Sink7

THROUGHP	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1059.90	8.05	1054.14	1065.66	1053.00	1079.00

##### Cumulative average : Sink7, THROUGPUT



## Overall results

SCENARIO	WEEK PRODUCTION
STANDARD	95,98
REDUCED REPAIR	109,39
BUFFFER BEFORE C AND D	105,99

## Use Case 2

### Assignment 1

Analyse each strategy accurately. On the basis of (theoretical) utilizations, try to estimate which system will operate the most efficiently.

My estimation will be that strategy 5 will operate the most efficient based on (theoretical) utilizations, because it won't put valuable resources to use when it's not completely necessary. All the other strategy's make use of a fixed personnel utilization.

The strategy to change the service time on a per item basis won't do allot in my estimation, because based on the probability distribution of "number of items per shopper" that was given. The average service time per customer, only on item basis, will roughly be the same as the service not on item basis. And when you add the payment time, I estimate the time will be a bit longer.

## Assignment 2

Build simulation models for all strategies and define for each the average queue and waiting time (in strategy 3 for both customer types).

checkouts open on Saturday from 9:00 to 17:00  
 average 1400 shoppers in a day (random distribution)

To write the correct 4dScript for the Interval-arrival Time in the Source Node I must calculate the average time between the shoppers. To do this I get the total time the shop is open and divide it by the number of shoppers. To get the random distribution of shoppers during the day I will use the negative Exponential over the average Interval-arrival Time between the shoppers.

The shop is open for 8 hours. That is a total of  $3600 * 8$  seconds = 28800 seconds

Now I will divide the open hours by the number of shoppers

$28800 / 1400 \approx 20,57$  seconds

To determine the best stochastic distribution for modeling customer service, I conducted some research. I found that the Poisson distribution is widely used due to its memoryless property, meaning the probability of a new customer arriving is independent of the time since the last customer arrived. Additionally, the Poisson distribution maintains a consistent average arrival rate, making it a reliable choice for such modeling.

The final 4dScript for the Interval-arrival Time will be “Poisson(20,57)”.

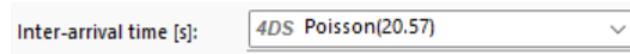


Figure 2 - Source node

## Strategy 1

(fixed personnel, service time independent on the number of groceries)

In this case, all checkouts are open and the service time for each customer is taken from a negative exponential distribution with an average of 2.5 minutes

*layout*

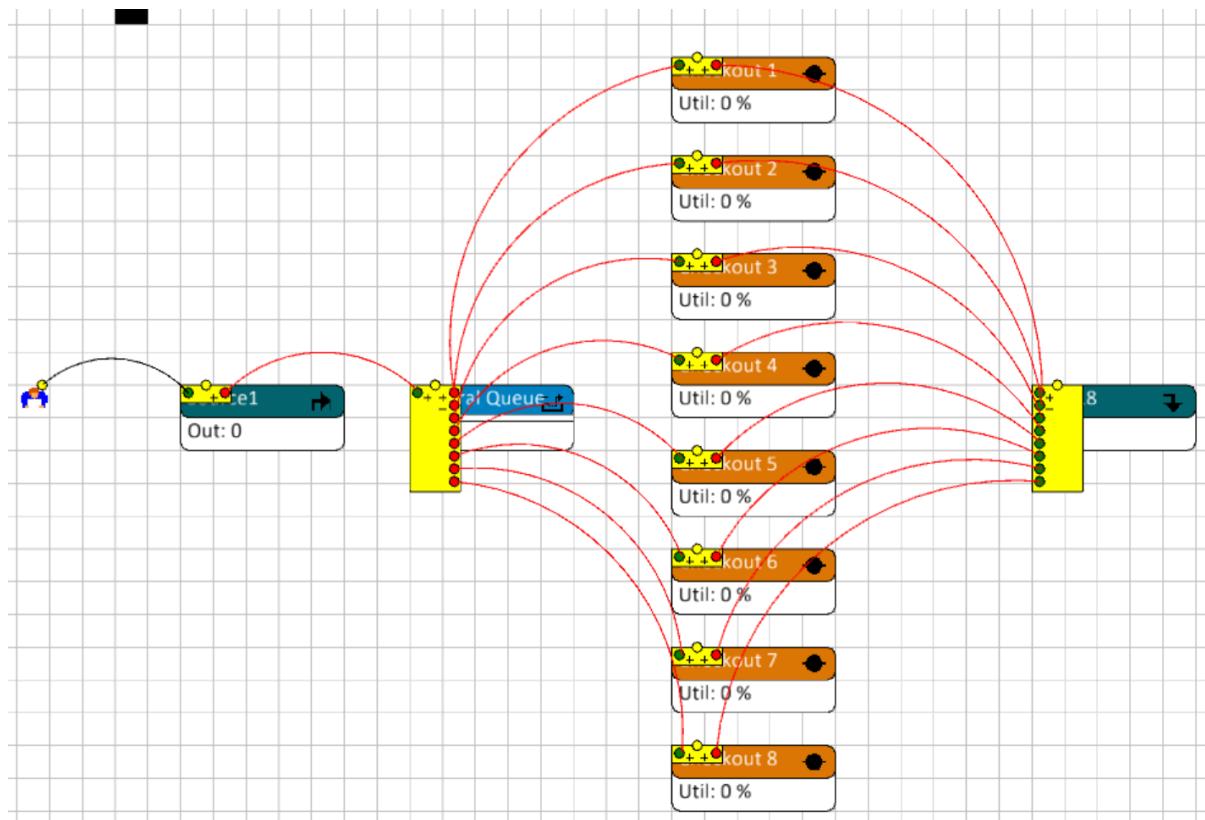


Figure 1 - strategy 1 layout

Settings	
Setup time [s]:	0
Cycletime [s]:	4DS negexp(mins(2.5))
Send to:	1
Input strategy:	1. Any inputchannel - First channel filled

Figure 3 - Checkout node

### *Experiment information*

Model:	supermarket1.mod.bak
Observation period:	288000
Warmup period:	0
Number of replications:	10
Simulation method:	Separate runs
Description:	

#### **1\_Entrance**

	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1397.30	8.58	1391.16	1403.44	1383.00	1414.00

#### **2\_Main Queue**

	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
<b>AvgContent</b>	3.31	2.12	1.80	4.83	0.98	8.10
<b>AvgStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	68.19	43.95	36.75	99.63	19.60	167.70
<b>MaxContent</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	18.40	6.52	13.74	23.06	11.00	29.00
<b>MaxStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	365.05	134.85	268.59	461.52	186.95	603.28

#### **3\_Checkouts Elements : Checkout 1, Checkout 2, Checkout 3, Checkout 4, Checkout 5**

	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
<b>AvgStay</b>	149.20	4.46	146.01	152.39	145.45	160.29
<b>MaxStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1138.02	142.07	1036.39	1239.65	979.38	1427.92

#### **4\_Exit**

	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
<b>Exited</b>	1387.10	10.71	1379.44	1394.76	1374.00	1409.00

Figure 2 - Experiment info strategy 1

	Avg Checkout Stay (seconds)	Avg Queue length (Customers)	Avg Queue Stay (seconds)
Strategy 1	149.20	3.31	68.19

Table 1 - info strategy 1

## Strategy 2

(fixed personnel, service time dependent on the number of groceries)

This is a variation of the first strategy where the service time depends on the number of items purchased. We assume that the scanning time per item is 6 seconds, and that payment takes 30 seconds.

Measurements have shown that the following empirical probability distribution can be used for the number of groceries:

<b>Number Of items</b>	<b>Probability (in %)</b>	<b>Number Of items</b>	<b>Probability (in %)</b>	<b>Number Of items</b>	<b>Probability (in %)</b>
1	2	18	3	35	1
2	2	19	3	36	1
3	2	20	3	37	1
4	2	21	3	38	1
5	2	22	3	39	1
6	3	23	3	40	1
7	3	24	3	41	1
8	3	25	3	42	1
9	3	26	2	43	1
10	3	27	2	44	1
11	3	28	2	45	1
12	3	29	2	46	1
13	3	30	2	47	1
14	3	31	1	48	1
15	3	32	1	49	1
16	3	33	1	50	1
17	3	34	1		

Figure 3 - probability distribution strategy 4

### Layout

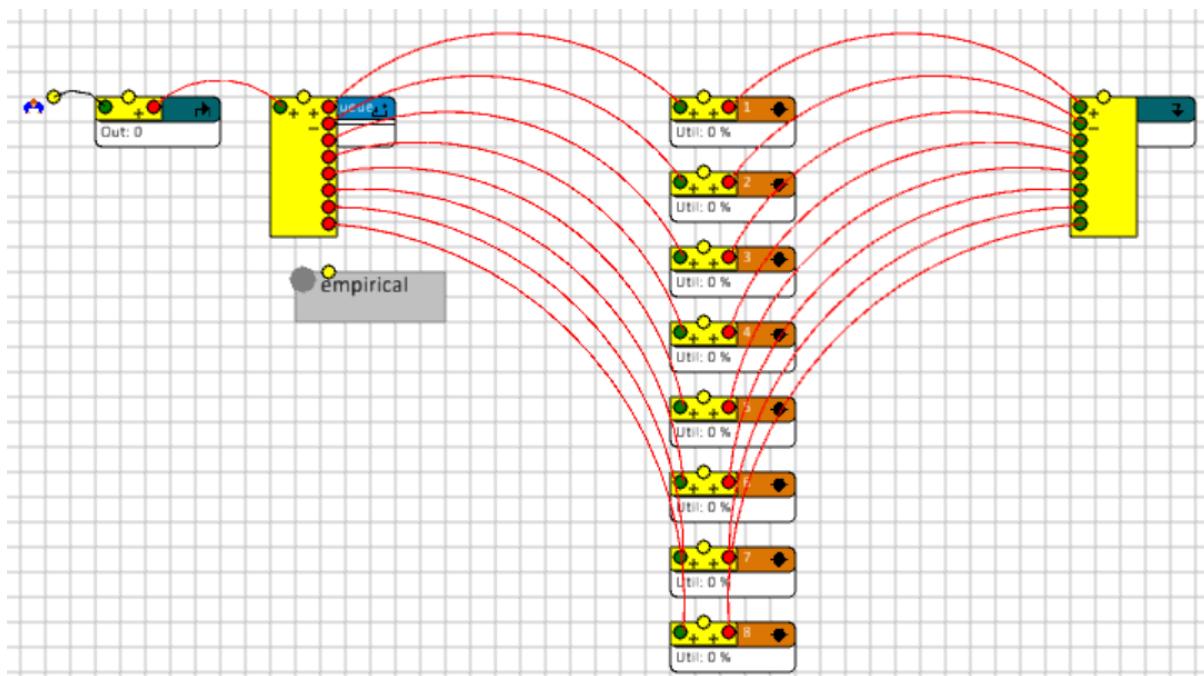
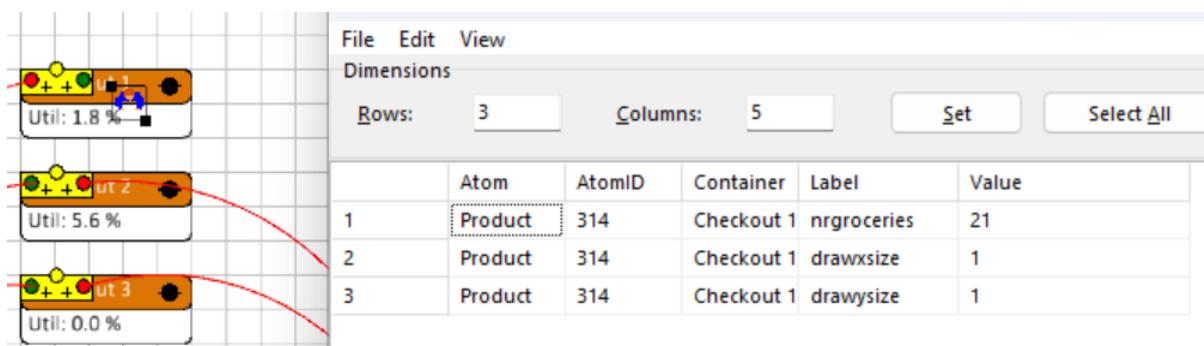


Figure 4 - strategy 2 layout



Dimensions

Rows: 3    Columns: 5    Set    Select All

	Atom	AtomID	Container	Label	Value
1	Product	314	Checkout 1	nrgroceries	21
2	Product	314	Checkout 1	drawxsize	1
3	Product	314	Checkout 1	drawysize	1

Figure 5 - Label Number of groceries on atom

Cycletime [s]:

Figure 6 - server cycletime [s]

### Experiment information

Model:	supermarket1_st2.mod.bak.mod
Observation period:	288000
Warmup period:	0
Number of replications:	10
Simulation method:	Separate runs
Description:	

### 1\_Entrance

Entered	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1397.90	7.50	1392.53	1403.27	1386.00	1409.00

### 2\_Main Queue

AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1.33	0.42	1.03	1.63	0.74	1.93
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	27.38	8.52	21.29	33.48	15.36	39.47
MaxContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	8.80	1.40	7.80	9.80	7.00	12.00
MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	165.50	29.83	144.16	186.84	122.00	210.00

### 3\_Checkouts Elements : Checkout 1, Checkout

AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	153.33	2.76	151.36	155.31	147.57	156.90
MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	330.00	0.00	330.00	330.00	330.00	330.00

### 4\_Exit

Exited	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1389.40	6.40	1384.82	1393.98	1375.00	1397.00

Figure 7 - Experiment info strategy 2

	Avg Checkout Stay (seconds)	Avg Queue length (Customers)	Avg Queue Stay (seconds)
Strategy 2	153.33	1.33	27.38

Table 2 - info strategy 2

### Strategy 3

(fixed personnel, service time dependent on the number of groceries and separation of customer types)

In this strategy, the first checkout is an express lane: only customers with 10 items or less are allowed to use this checkout. In this case, we assume that all these customers will actually use the express lane

#### Layout

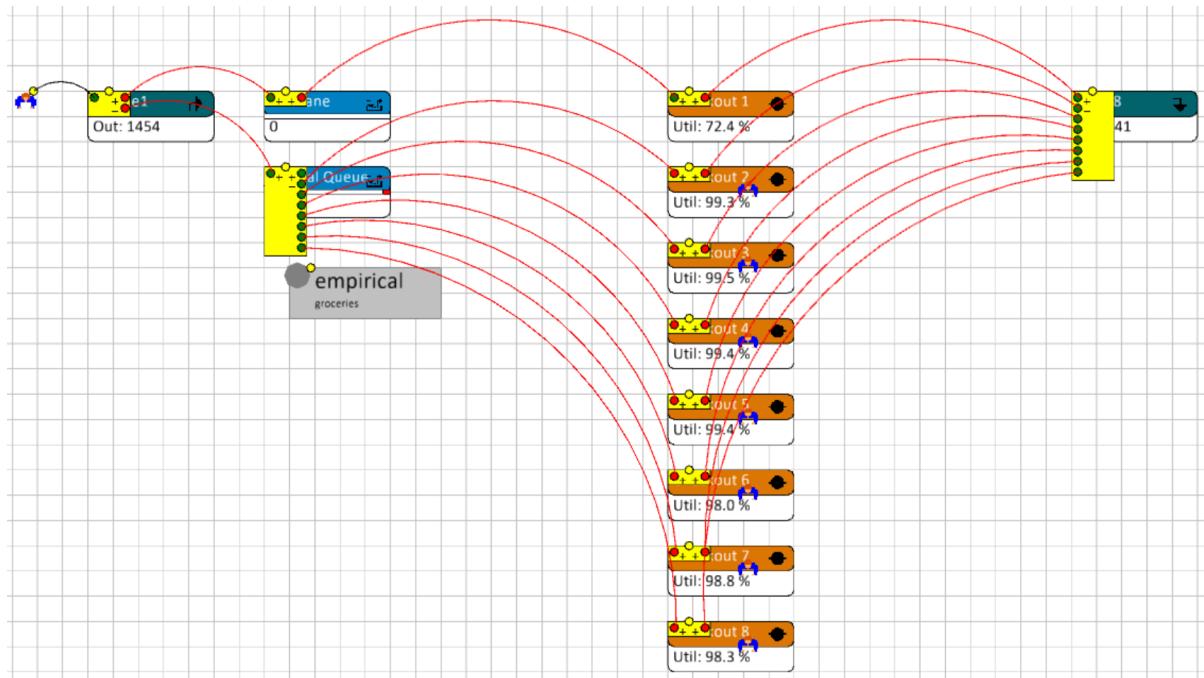


Figure 8 - Strategy 3 layout

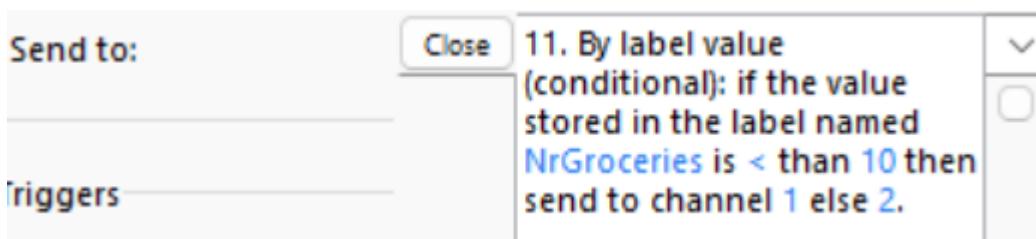


Figure 9 - Source send conditional

## Experiment information

Model:	supermarket1_st3.mod						Model:	supermarket1_st3.mod					
Observation period:	288000						Observation period:	288000					
Warmup period:	0						Warmup period:	0					
Number of replications:	10						Number of replications:	10					
Simulation method:	Separate runs						Simulation method:	Separate runs					
Description:							Description:						
<b>1_Entrance</b>							<b>1_Entrance</b>						
Entered	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	Entered	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1401.80	8.18	1395.95	1407.65	1392.00	1418.00		1404.20	7.38	1398.92	1409.48	1394.00	1417.00
<b>2_Fast lane</b>							<b>2_Fast lane</b>						
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	43.63	11.94	35.09	52.17	30.50	70.04		5630.71	1253.63	4733.91	6527.50	4083.70	7839.38
Total	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	Total	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	305.90	10.19	298.61	313.19	285.00	319.00		188.70	11.04	180.81	196.59	171.00	206.00
<b>2_Main Queue</b>							<b>2_Main Queue</b>						
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	104.24	54.40	65.33	143.15	31.86	187.26		28.62	13.34	19.08	38.16	14.75	51.95
Total	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	Total	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1092.30	8.92	1085.92	1098.68	1078.00	1104.00		1089.40	14.90	1078.74	1100.06	1058.00	1114.00
<b>2_Queue Elements : 2_Fast lane, 2_Main Queue</b>							<b>2_Queue Elements : 2_Fast lane, 2_Main Queue</b>						
AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	4.44	2.11	2.93	5.95	1.60	7.73		62.65	14.09	52.57	72.72	47.06	89.16
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	91.18	43.08	60.37	122.00	32.98	156.17		852.45	181.26	722.78	982.11	680.67	1233.06
MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	397.40	137.20	299.26	495.54	244.00	697.00		11342.99	1754.01	10088.24	12597.73	9357.20	13741.67
<b>3_Checkouts Elements : Checkout 1, Checkout</b>							<b>3_Checkouts Elements : Checkout 1, Checkout</b>						
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	152.34	1.71	151.11	153.56	149.43	154.89		151.43	3.59	148.86	154.00	147.83	159.14
MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	330.00	0.00	330.00	330.00	330.00	330.00		1141.50	106.37	1065.41	1217.59	994.42	1295.83
<b>4_Exit</b>							<b>4_Exit</b>						
Exited	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum	Exited	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1390.70	7.42	1385.39	1396.01	1381.00	1404.00		1271.30	18.69	1257.93	1284.67	1246.00	1303.00

Figure 11 - Experiment info strategy 3 – Groceries Count

Figure 10 - Experiment info strategy 3 - Exp timing

	Avg Checkout Stay (seconds)	Avg Queue length (Customers)	Avg Queue Stay (seconds)
Strategy 3 Groceries-Count*	152.34	4.44	91.18
Strategy 3 Exp-Timing*	151.43	62.65	852.45

Table 3 - info strategy 3

\*Groceries-Count – The checkout time is based on the number of groceries

\*Exp-Timing – the checkout time is based on negative exponential distribution

## Strategy 4

(fixed personnel, service time dependent on the number of groceries, with priority to the customers with the least groceries)

This alternative is interesting from the point of view of flows, but would lead to major problems in supermarkets. Modify the second model so that customers with the fewest items are served first

### Layout

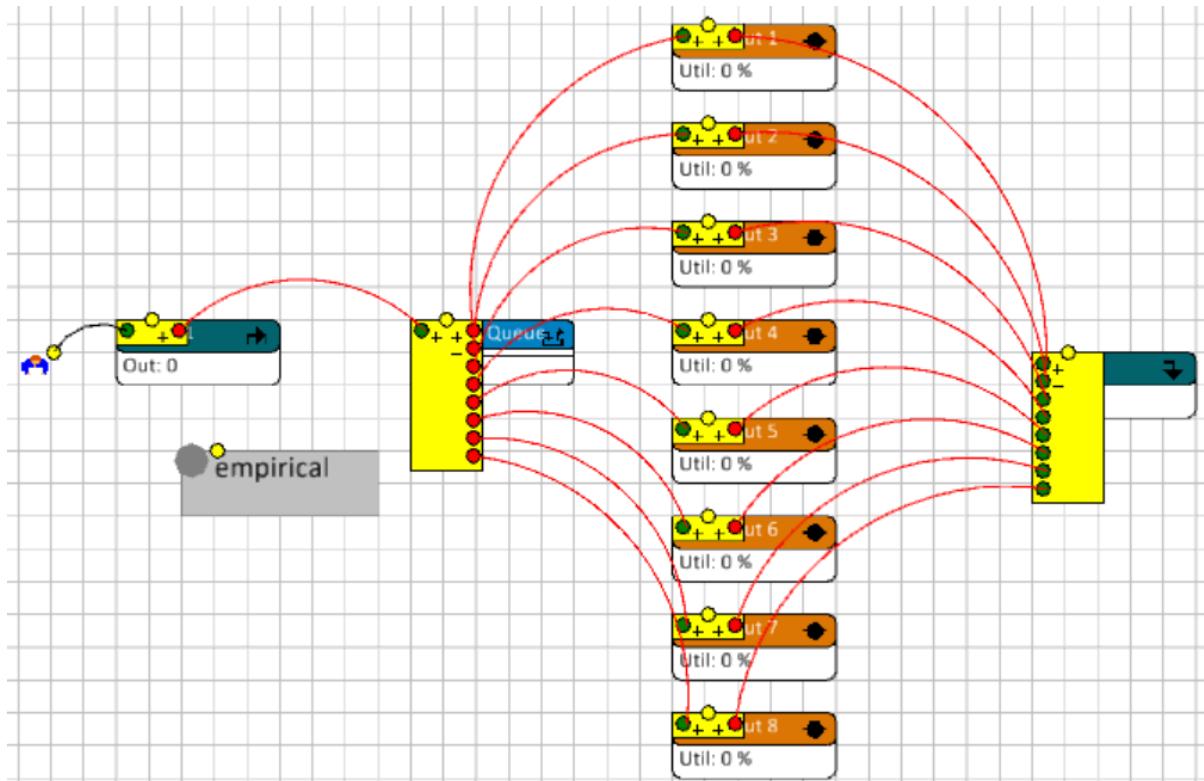
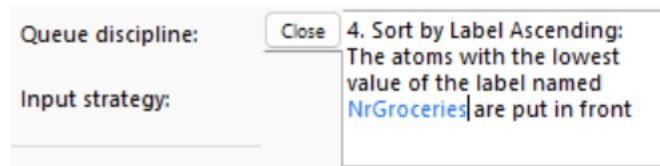


Figure 12 - strategy 4 layout



### *Experiment information*

Model: supermarket1\_st4.mod  
 Observation period: 288000  
 Warmup period: 0  
 Number of replications: 10  
 Simulation method: Separate runs  
 Description:

#### **1\_Entrance**

<b>Entered</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1404.40	7.47	1399.06	1409.74	1388.00	1414.00

#### **2\_Main Queue**

<b>AvgContent</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1.03	0.26	0.84	1.21	0.68	1.59
<b>AvgStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	20.99	5.25	17.23	24.75	13.91	32.51
<b>MaxContent</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	6.70	0.82	6.11	7.29	6.00	8.00
<b>MaxStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	830.50	378.62	559.65	1101.35	359.00	1374.00

#### **3\_Checkouts Elements : Checkout 1, Checkout 2, Checkout 3, Checkout 4, Checkout 5**

<b>AvgStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	153.55	1.76	152.29	154.81	150.83	156.55
<b>MaxStay</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	330.00	0.00	330.00	330.00	330.00	330.00

#### **4\_Exit**

<b>Exited</b>	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1395.30	6.88	1390.38	1400.22	1381.00	1404.00

Figure 13 - Experiment info strategy 4

	Avg Checkout Stay (seconds)	Avg Queue length (Customers)	Avg Queue Stay (seconds)
Strategy 4	153.55	1.03	20.99

Table 4 - info strategy 4

## Strategy 5

(variable personnel, service time dependent on the number of people waiting)

The availability of checkouts is now linked to the number of customers in the queue: if 0 to 3 customers are waiting, checkout 1 opens, 4 to 6 customers, checkout 2 etc. up to a total of 8 checkouts for 22 customers or more. When the number of customers queuing decreases, the corresponding number of checkouts closes

### Layout

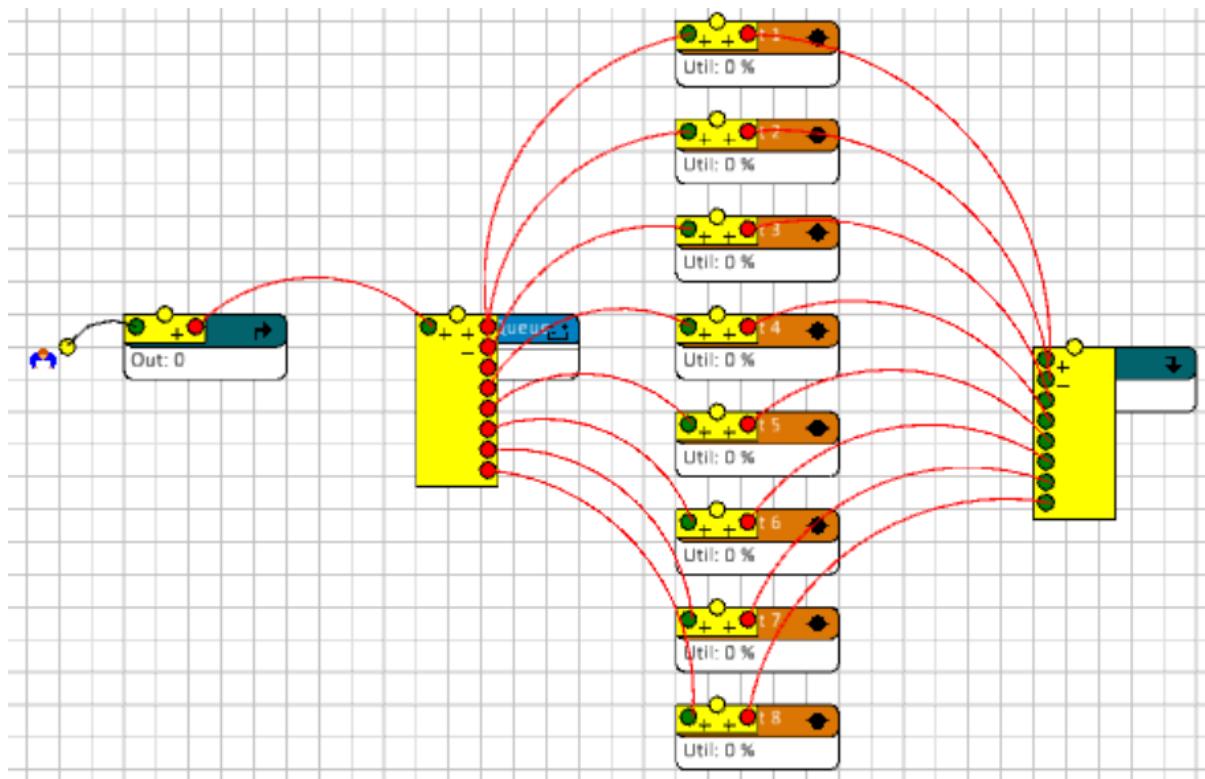


Figure 14 - Strategy 5 layout

```

do (
    if(Content(c) > 21, OpenInput(out(1,c)), CloseInput(out(1,c))),
    if(Content(c) > 18, OpenInput(out(2,c)), CloseInput(out(2,c))),
    if(Content(c) > 15, OpenInput(out(3,c)), CloseInput(out(3,c))),
    if(Content(c) > 12, OpenInput(out(4,c)), CloseInput(out(4,c))),
    if(Content(c) > 9, OpenInput(out(5,c)), CloseInput(out(5,c))),
    if(Content(c) > 6, OpenInput(out(6,c)), CloseInput(out(6,c))),
    if(Content(c) > 3, OpenInput(out(7,c)), CloseInput(out(7,c))),
    if(Content(c) > 0, OpenInput(out(8,c)), CloseInput(out(8,c)))
)

```

Figure 15 - 4dScript queue entry and exit



Figure 16 - Trigger settings queue

### Experiment information

Model:	supermarket1_st5.mod
Observation period:	288000
Warmup period:	0
Number of replications:	10
Simulation method:	Separate runs
Description:	

### 1\_Entrence

Entered	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1394.70	7.35	1389.44	1399.96	1387.00	1410.00

### 2\_Main Queue

AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	20.38	1.04	19.63	21.12	19.11	22.58
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	423.49	23.05	407.00	439.98	398.19	471.48
MaxContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	35.80	3.05	33.62	37.98	32.00	40.00
MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	726.66	62.80	681.74	771.59	641.42	818.09

### 3\_Checkouts Elements : Checkout 1, Checkout 2, Checkout 3, Checkout 4, Checkout 5

AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	150.85	3.31	148.48	153.22	145.92	155.46
MaxStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1148.78	205.82	1001.55	1296.01	955.82	1508.91

### 4\_Exit

Exited	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1363.20	12.68	1354.13	1372.27	1339.00	1385.00

Figure 17 - Experiment info strategy 5

	Avg Checkout Stay (seconds)	Avg Queue length (Customers)	Avg Queue Stay (seconds)
Strategy 5	150.85	20.38	423.49

Table 5 - info strategy 5

## Results

	Strategy 1	Strategy 2	Strategy 3		Strategy 4	Strategy 5
			Groc-Count	Exp-Time		
<i>Queues</i>						
<b>Avg Content (customers)</b>	3.31	1.33	4.44	62.65	1.03	20.38
<b>Avg Stay (seconds)</b>	68.19	27.38	91.18	852.45	20.99	423.49
<b>Max Content (customers)</b>	18.40	8.80	/	/	6.70	35.80
<b>Max Stay (seconds)</b>	365.05	165.50	397.40	11342.99	830.50	726.66
<i>Checkouts</i>						
<b>Avg Stay (seconds)</b>	149.20	153.33	152.34	151.43	153.55	150.85
<b>Max Stay (seconds)</b>	1387.10	330.00	330.00	1141.50	330.00	1148.78

Strategy Explained*	
Strategy 1	<ul style="list-style-type: none"> <li>server time negExp(min(2.5))</li> </ul>
Strategy 2	<ul style="list-style-type: none"> <li>server time depends on amount of items item = 6 seconds payment = 30 seconds</li> </ul>
Strategy 3	<ul style="list-style-type: none"> <li>service time depends on amount of items item = 6 seconds payment = 30 seconds</li> <li>express lane for items <math>\leq 10</math></li> </ul>
Strategy 4	<ul style="list-style-type: none"> <li>service time depends on amount of items item = 6 seconds payment = 30 seconds</li> <li>least items first out</li> </ul>
Strategy 5	<ul style="list-style-type: none"> <li>server time negExp(min(2.5))</li> <li>variable checkouts           <ul style="list-style-type: none"> <li>1 open = <math>0 &lt; \text{customers} \leq 3</math></li> <li>2 open = <math>3 &lt; \text{customers} \leq 6</math></li> <li>3 open = <math>6 &lt; \text{customers} \leq 9</math></li> <li>4 open = <math>9 &lt; \text{customers} \leq 12</math></li> <li>5 open = <math>12 &lt; \text{customers} \leq 15</math></li> <li>6 open = <math>15 &lt; \text{customers} \leq 18</math></li> <li>7 open = <math>18 &lt; \text{customers} \leq 21</math></li> <li>8 open = <math>21 &lt; \text{customers}</math></li> </ul> </li> </ul>

\*

- 8 checkout lanes
- checkouts open on Saturday from 9:00 to 17:00
- average 1400 shoppers in a day (random)
- fifo
- checkout operators work with the same speed

(I use poisson distribution for the random distribution of shoppers)

## Assignment 3

Which system would you recommend? Can they be compared with each other? Are there better alternatives?

When I look at the results of the different strategies, I deduct that the different strategies in service time is negligible in the case of average service time. But the standard deviation is considerably higher when using the negative exponential distribution compared to the empirical distribution with the number of items. I assume that these higher deviations have a correlation with the extended average staying times in the queue.

I would recommend using the empirical distribution in case of service times.

Looking at the different strategies with the use of the empirical distribution. I deduct that strategy 3 has a significant higher queueing time compared to strategy 2 and strategy 4. This make me want to rule out strategy 3 and the use of a fast lane, because I don't see a significant benefit from it.

Strategy 4, which prioritizes customers with fewer items by placing them at the front of the queue, results in the shortest average queueing time. However, it also produces significant outliers. In contrast, Strategy 2, which follows a first-in, first-out (FIFO) approach, is more consistent, showing a substantial reduction in outliers.

I recommend using strategy 2. It doesn't significantly punish the queueing time of the customer on the number of items bought and it is very stable. If the punishment of customers with a larger number of groceries is not a problem, then go for strategy 4 with the lower overall stay time.

## Conclusion

<b>Variant</b>	<b>Queue</b>	<b>Waiting time (in seconds)</b>
1	6.8 ± 1.0	139 ± 20
2	5.5 ± 0.8	112 ± 15
3 (customers with 10 or less items)	1.6 ± 0.2	129 ± 13
3 (customers with more than 10 items)	6.3 ± 1.2	171 ± 30
4	4.2 ± 0.6	85 ± 12
5	22.3 ± 0.8	458 ± 13

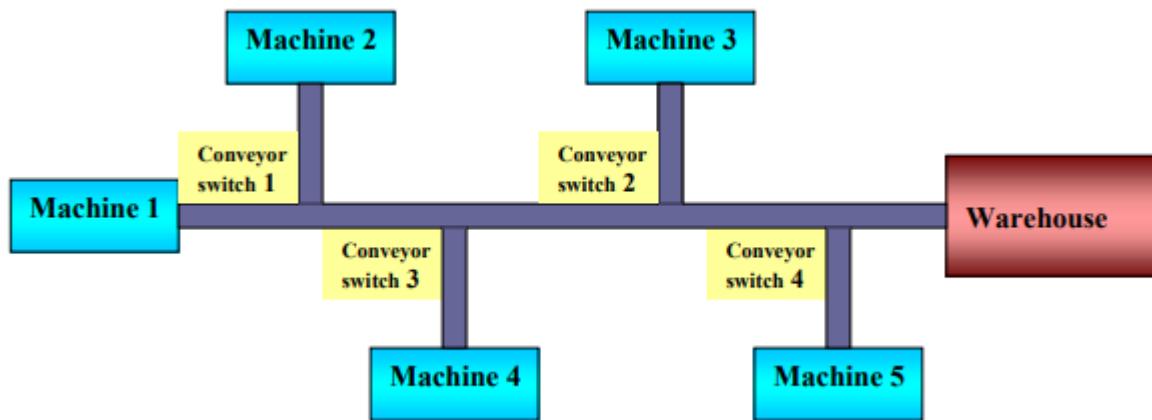
Figure 18 - Answer results

I notice that the results of the given answers are very different compared to the result I found. Only the results from strategy 5 are comparable.

I found that the reason for this difference is the stochastic distribution that is used for the Inter-arrival time. When I change the inter-arrival time to a negative exponential distribution the resulted values are similar.

I used the poisson distribution for the simulation, because the simulated process is a M/M/c queueing model. And I found that this is the most commonly used stochastic distribution for this type of queueing model, mainly because of its memoryless property.

## Use Case 3



## Assignment 1

Make your own calculation / reasoned estimate beforehand regarding the transport system capacity in the case of an equal production per machine (in boxes per hour). Where will the system first jam?

In my capacity estimates for the system, I'll be disregarding potential jams due to the complexity of calculating them accurately.

Each machine produces 4 products per minute, which equals 240 boxes per hour. Multiplying this by the number of machines gives us the theoretical maximum production rate: 1200 boxes per hour for the entire system.

Process time:

- Machine line 1 : 0 sec
- machine line 2 : 2 sec
- machine line 3 : 2 sec
- machine line 4 ; 10 sec
- machine line 5 : 5 sec

The longest process time occurs at Conveyor Switch 3, where boxes are most likely to accumulate due to the delay. This makes Conveyor Switch 2 the likely point for jams to occur.

## Assignment 2

Determine with simulation if a production of 4 boxes per minute per machine is possible and the maximum feasible system capacity if all machines generate the same production.

Looking at the experiment results under “Experiment” you can see that the standard deviation for both the average content and average stay on each conveyor belt is either 0 or 0.01. Which I assume would mean that there are no concerning product jams that should be taken care off.

I expect that the maximum feasible system capacity will be around 6 boxes per minute. The way I got that assumption is because of machine line 4 it's transfer time. It has a transfer time of 10 sec which, I assume, should mean that if the production of that line goes above that it will jam. But apparently when I create a Uniform distribution

layout

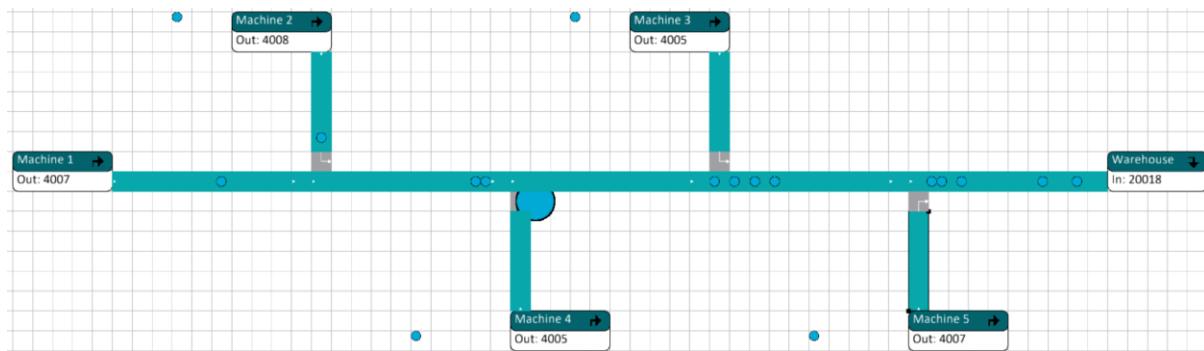


Figure 19 - transport system layout

```
Triggers
1 Do (
2  CloseInput(out(2,c)),
3  LoopUntil(Content(Central(1,c)) = 0, Break)
4 )
```

Figure 20 - Corner Transfer Unit, Entry trigger 4dscript

```
Trigger on exit: 1 OpenInput(out(2,c))
```

Figure 22- Corner Transfer Unit, Exit trigger 4dscript

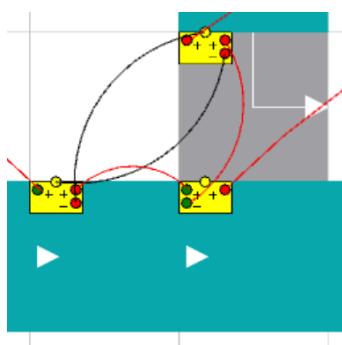


Figure 21 - Intersection channel connections

## Experimental information

### Experiment information

Model: transport1.mod  
 Observation period: 288000  
 Warmup period: 0  
 Number of replications: 10  
 Simulation method: Separate runs  
 Description:

### Machines Elements : Machine 1, Machine 2,

Machine	Production	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	9597.00	2.40		9595.28	9598.72	9593.00	9601.00

### Main conveyor Elements : Accumulating

Machine	AVGCONTE	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AVGSTAY	10.66	0.08	10.60	10.72	10.53	10.81
	AVGSTAY	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 1

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	0.33	0.00	0.33	0.33	0.33	0.33
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 2

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	0.33	0.00	0.33	0.33	0.33	0.33
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 3

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	0.33	0.00	0.33	0.33	0.33	0.33
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 4

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	0.33	0.00	0.33	0.33	0.33	0.33
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Warehouse

Output	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	9583.10	2.69	9581.18	9585.02	9578.00	9586.00

Model: transport1.mod

Observation period: 288000

Warmup period: 0

Number of replications: 10

Simulation method: Separate runs

Description:

### Machines Elements : Machine 1, Machine 2,

Machine	Production	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
		11299.70	2.91	11297.62	11301.78	11297.00	11305.00

### Main conveyor Elements : Accumulating

Machine	AVGCONTE	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AVGSTAY	45.37	0.02	45.35	45.38	45.33	45.39
	AVGSTAY	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 1

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	9.82	0.00	9.82	9.83	9.82	9.83
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 2

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	0.50	0.00	0.50	0.50	0.50	0.50
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 3

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	9.66	0.02	9.65	9.68	9.63	9.69
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Side Accumulating Conveyor 4

Machine	AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	AvgStay	0.50	0.00	0.50	0.50	0.50	0.50
	AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum

### Warehouse

Output	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	11230.40	2.91	11228.32	11232.48	11228.00	11236.00

Figure 24 - Experiment info Uniform(14.5 ,15.5)

Figure 23 - Experiment info Uniform(9.5,10.5)

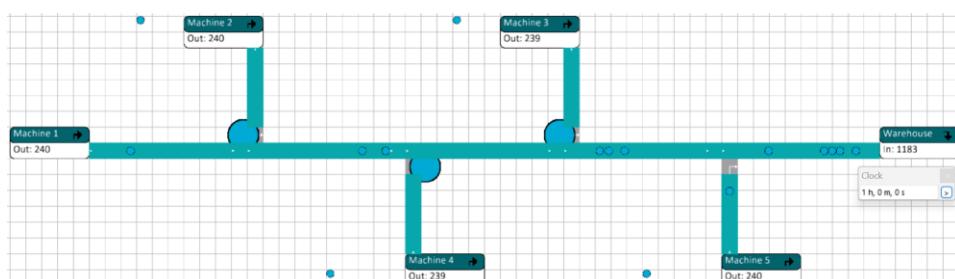


Figure 25 - Experiment Uniform (14.5,15.5)

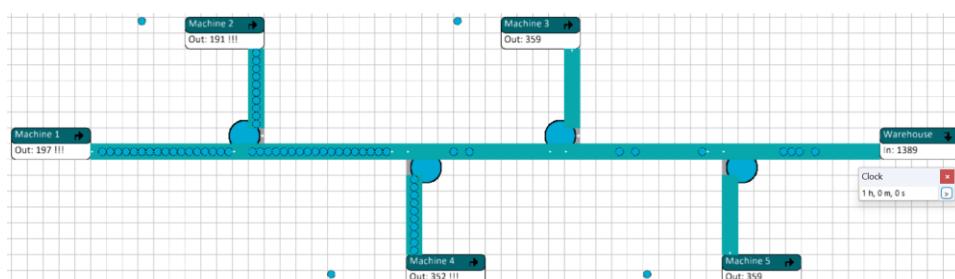


Figure 26 - Exirment Uniform (9.5,10.5)

Model: transport1.mod  
 Observation period: 288000  
 Warmup period: 0  
 Number of replications: 10  
 Simulation method: Separate runs  
 Description:

#### Machines Elements : Machine 1, Machine 2,

Production	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	13331.20	3.74	13328.53	13333.87	13324.00	13338.00

#### Main conveyor Elements : Accumulating

AVGCONTE	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	18.29	1.00	17.57	19.00	17.04	20.43
AVGSTAY	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	7.91	0.43	7.60	8.22	7.37	8.83

#### Side Accumulating Conveyor 1

AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	0.47	0.00	0.46	0.47	0.46	0.48
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	5.02	0.05	4.99	5.06	5.00	5.16

#### Side Accumulating Conveyor 2

AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	0.46	0.00	0.46	0.46	0.46	0.46
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	5.00	0.00	5.00	5.00	5.00	5.00

#### Side Accumulating Conveyor 3

AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	0.48	0.00	0.48	0.48	0.48	0.49
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	5.22	0.01	5.22	5.23	5.21	5.24

#### Side Accumulating Conveyor 4

AvgContent	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	0.46	0.00	0.46	0.46	0.46	0.46
AvgStay	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	5.00	0.00	5.00	5.00	5.00	5.00

Warehouse Output	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	13309.20	6.66	13304.43	13313.97	13295.00	13320.00

Figure 27 - Experiment info Uniform (10.3,11.3)

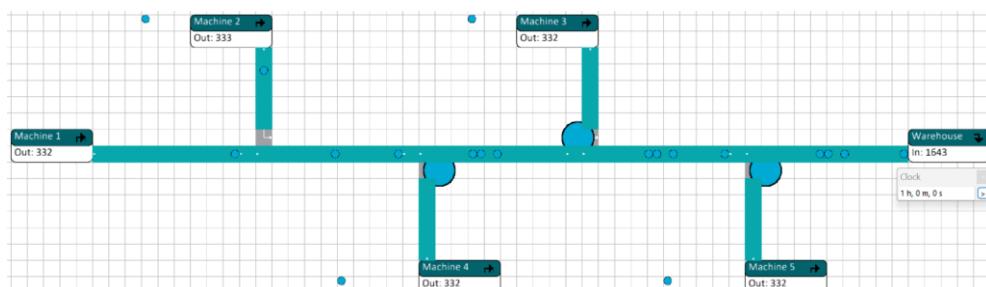


Figure 28 - Experiment Uniform (10.3,11.3)

## Results

The maximum feasible system capacity if all machines generate the same production seems to be around the uniform distribution of (10.3, 11.3). Anything lower will result in a product jam beginning at the intersection with machine line 4.

This is a machine production of around 5.56 boxes per minute.

## Assignment 3

Can the total system production be increased by varying the production quantity per machine? Investigate this by means of simulation. Warning! The maximum technical capacity of the machines is 8 boxes per minute.

### Experiment information

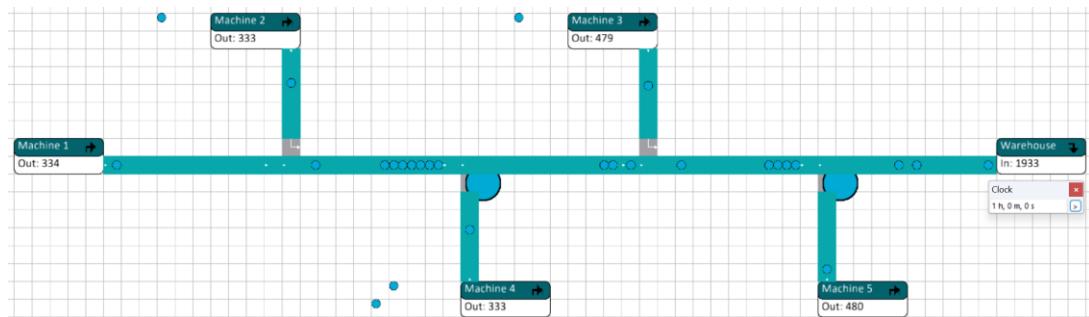


Figure 29 - Experiment variable production

Machine nr.	Inter-Arrival time	Δ Box per min
Machine 1	Uniform(10.3, 11.3)	5,56
Machine 2	Uniform(10.3, 11.3)	5,56
Machine 3	Uniform(10.3, 11.3)	5,56
Machine 4	Uniform(7, 8)	8
Machine 5	Uniform(7, 8)	8

Table 6 - variable production machines

### Result

In “Table 6 - variable production machines” the minimum interval-arrival time without the stalling of machines can be seen. The production can be increased more, but this will go against the machine utilization. The machines will stall because of product jams on the conveyor belt.

I notice that the front 2 side conveyors don't have any problems with product jams even when machine 3 and machine 5 are at maximum production capacity. The main congestion starts to happen the conveyor switch 2 (machine line 4), because of its high transfer time compared to the rest of machine lines.

## Assignment 4

What are your suggestions in order to improve the system? Does increasing the speed of the conveyors produce an effect?

### Experimental information

#### Increased conveyor speed

The conveyor speed is increased from 1 m/s to 2 m/s.

Machine nr.	Inter-Arrival time	Δ Box per min
Machine 1	Uniform(7, 8)	8
Machine 2	Uniform(7, 8)	8
Machine 3	Uniform(10.3, 11.3)	5,56
Machine 4	Uniform(7, 8)	8
Machine 5	Uniform(7, 8)	8

Table 7 - Variable production increased conveyor speed

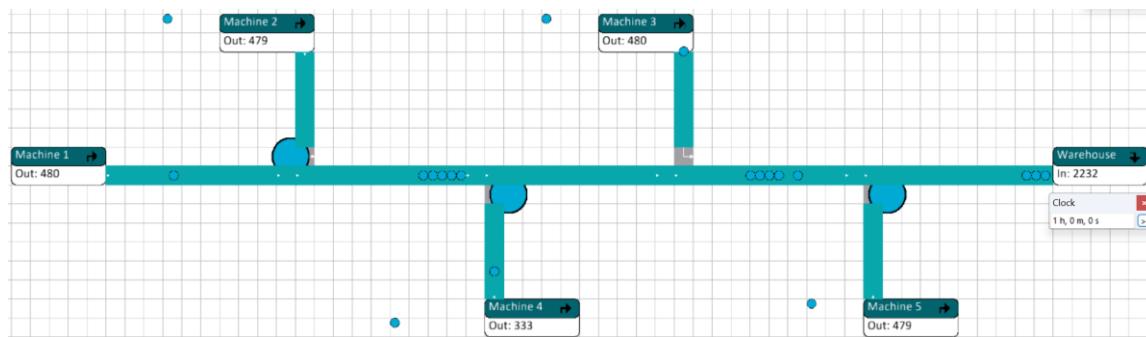


Figure 30 - Experiment variable production increased conveyor speed

Model: transport3.mod  
 Observation period: 288000  
 Warmup period: 0  
 Number of replications: 10  
 Simulation method: Separate runs  
 Description:

Machines Elements : Machine 1, Machine 2,						
Production	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	18024.80	4.73	18021.41	18028.19	18017.00	18031.00
Status Busy	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1.00	0.00	1.00	1.00	1.00	1.00

Figure 31 - Experiment info variable production increased conveyor speed

### Transfer unit switch

The transfer unit of machine 2 and machine 3 are switched.

Machine line nr.	Inter-Arrival time	$\Delta$ Box per min	Transfer-time (s)
Machine 1	Uniform(7, 8)	8	0
Machine 2	Uniform(10.3, 11.3)	5,56	10
Machine 3	Uniform(7, 8)	8	2
Machine 4	Uniform(7, 8)	8	2
Machine 5	Uniform(7, 8)	8	5

Table 8 - Variable production Transfer unit switch

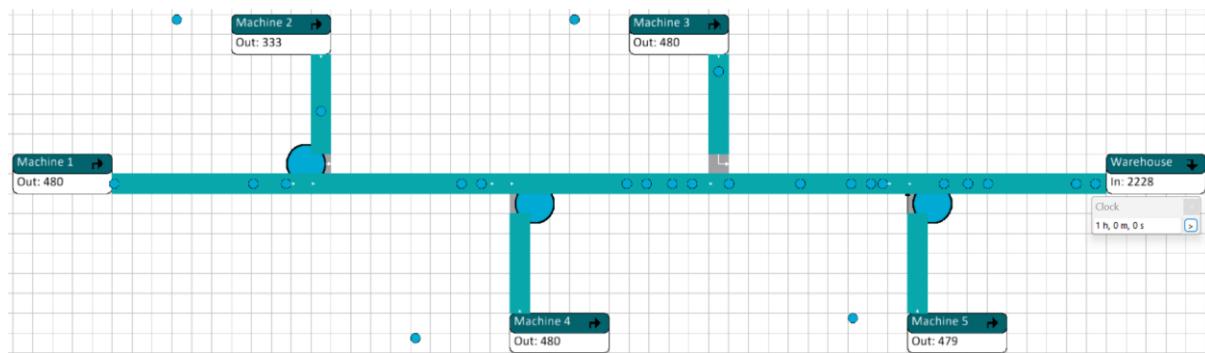


Figure 32 - Experiment variable production transfer unit switch

Model: transport3.mod  
 Observation period: 288000  
 Warmup period: 0  
 Number of replications: 10  
 Simulation method: Separate runs  
 Description:

### Machines Elements : Machine 1, Machine 2,

Production	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	18024.10	4.53	18020.86	18027.34	18017.00	18031.00
Status Busy	Average	S.Dev.	LB (95%)	UB (95%)	Minimum	Maximum
	1.00	0.00	1.00	1.00	1.00	1.00

Figure 33 - Experiment info variable production transfer unit switch

## Result

### *Increased conveyor speed*

The increase in conveyor speeds results in a higher capacity load for the transport line, without a loss of machine utilization. This can be seen in “Figure 31 - Experiment info variable production increased conveyor speed”. The busy status performance measure stays at 1.00 which means that there are no dips in utilization. Even when the machine production for machine 1 and machine 2 is set to its maximum production capability.

It is still important to leave a slight gap for machine 4 in production time compared to the transfer time. Otherwise, there will be no time for products to surpass the conveyor switch 3 and will create a product jam.

### *Transfer unit switch*

The change in transfer units for machine line 2 and machine line 4 has the same affect as the increase in conveyor speed. Machine 1 and machine 4 can be set to maximum capacity without it affecting any machine utilization. As can be seen in “Figure 33 - Experiment info variable production transfer unit switch“.

## Recommendation

I would recommend the transfer unit switch method with variable production quantity, because it will have the same benefits as the increased conveyor speed. And in this case, there is no increase in conveyer speed necessary. Which results in less power consumption compared to the other option.

## Conclusion

<b>Scenarios</b>	<b>Machine 1</b>	<b>Machine2</b>	<b>Machine 3</b>	<b>Machine 4</b>	<b>Machine 5</b>
<b>scenario 1</b>					
<b>boxes p/m</b>	4	4	4	4	4
<b>iat (in sec)</b>	15±½	15±½	15±½	15±½	15±½
<b>scenario 2</b>					
<b>boxes p/m</b>	5	5	5	5	5
<b>iat (in sec)</b>	12±½	12±½	12±½	12±½	12±½
<b>scenario 3</b>					
<b>boxes p/m</b>	8	8	4	8	8
<b>iat (in sec)</b>	7½±½	7½±½	15±½	7½±½	7½±½

Figure 34 - given results use case 3

I noticed that values are very similar between the given results (Figure 34 - given results use case 3) and my results. My results seem to have a little better result for scenario 3.

I also read that the recommendation and conclusion written in the results where very different compared to mine. The resulted conclusion was that speeding up the conveyor has no affect on the system. My results would say the opposite. I have not yet figured out why.

## Use Case 4

### Assignment 1

For the first assignment you are going to create a base model to work from. In the image below you can find a factory line containing multiple (different) machines. Copy this factory line into an Enterprise Dynamics model. Each station should only be connected to their closest neighboring station. Except for “De-paneling robot” and “Pin stitcher”, they should both be connected to “Assembly stage 1” where they will be combined into 1 product. “Start scanner” and “Board programming” will be the first machines after the “source(s)”. Do not forget to add a “Sink” after “Packaging and connector inspection.

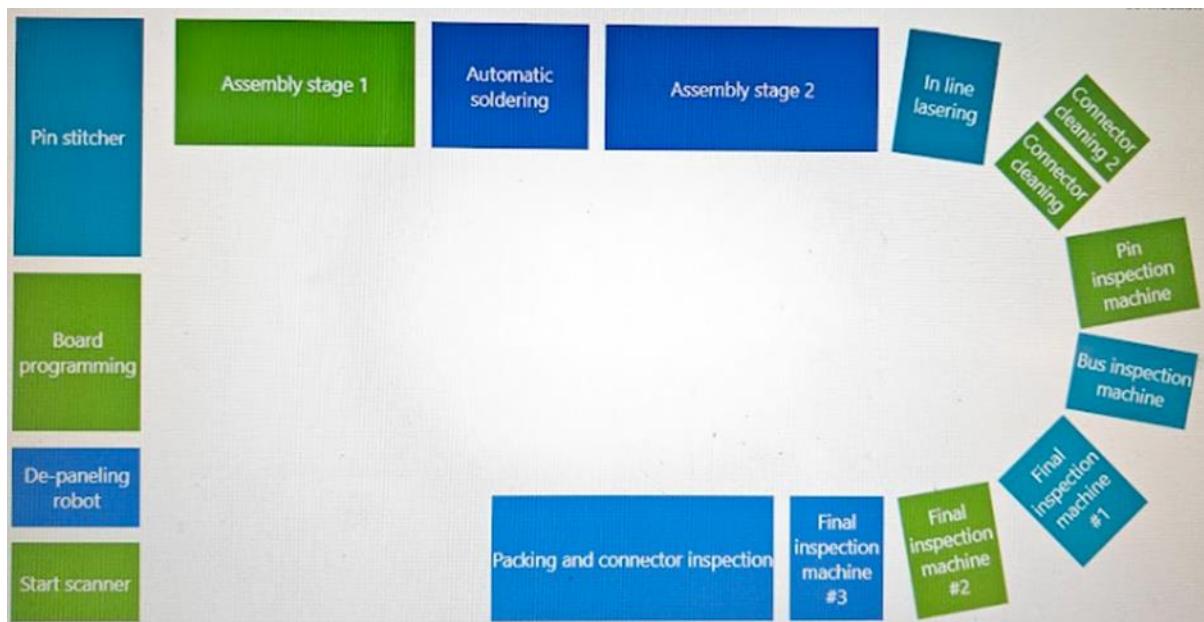


Figure 35 - Production line overview

Step	Task	Time it took	Timestamp	Operator
1	Grabs a circuit board	1,56	1,56	Operator 1
2	Puts the circuit board in the depaneling machine and turns it on	2,82	4,38	Operator 1
3	The depaneling machine is done	2:07,04	2:11,42	None
4	The pin stitcher machine is turned on	2:23,89	4:35,31	Operator 1
5	The circuit board gets taken out of the depaneling machine	47,75	5:23,06	Operator 1
6	The casing gets taken out of the pin stitcher and added together with the circuit board at assembly stage 1	24,93	5:47,99	Operator 1
7	The board gets put in the soldering machine	1,58	5:49,57	Operator 1
8	The board gets taken out of the soldering machine	57,69	6:47,26	Operator 1
9	The board gets put in a basket before assembly stage 2	10,40	6:57,66	Operator 1
10	Start the assembly at stage 2 using the beamer	26,07	7:23,73	Operator 1
11	Assembly at stage 2 is done and awaiting movement to the laser	17,18	7:40,91	None
12	The board gets put in the laser	34,27	8:15,18	Operator 1
13	The laser is done and waiting for operator pickup to cleaning	1:23,00	9:38,18	None
14	The board gets put in the cleaning for stage 1	20,64	9:58,82	Operator 2
15	The board gets turned around for cleaning stage 2	14,53	10:13,35	Operator 2
16	The board gets put into the pin inspection	55,62	11:08,97	Operator 2
17	The board gets put into the BUS checker	40,33	11:49,30	Operator 2
18	The board gets put into the final checker and the light turns red	1:23,88	13:13,18	Operator 2
19	The light turns green and the machine moves the board so it can be picked up	5,10	13:18,28	None
20	The board gets picked up and put in the basket at assembly line 3	38,55	13,56,83	Operator 2
21	The board gets put on a manual pin checking machine	18,66	14:15,49	Operator 2
22	The board gets checked	1,68	14:17,17	Operator 2
23	The board gets packaged into a plastic bag	44,00	15:01,17	Operator 2
24	The box is folded	7,15	15:08,32	Operator 2
25	The packaged board gets put into the box	2,43	15:10,75	Operator 2
26	The packaged board gets closed and wait for movement to the blue bin	30,69	15:41,44	Operator 2

Table 9 - Observation dataset

## Analysis

### Task analysis

Each station connected to closed neighbor

Exception for ... connected to assembly stage 1:

- Panneling robot
- Pin stitcher

'Source' before 'Start scanner' and 'Board programming'

'Sink' after 'Packaging and connector inspection'

To get a better understanding of how the simulation should work I looked at the observation data set inside ObservationDataSet.xlsx document (Table 9 - Observation dataset). This dataset gives some extra information about the product transfers between the systems. It suggests that certain transfers and system starts are handled by different operators. This means that additional parts need to be considered in this process. Like transfer by operators

## Design

### Prototypes

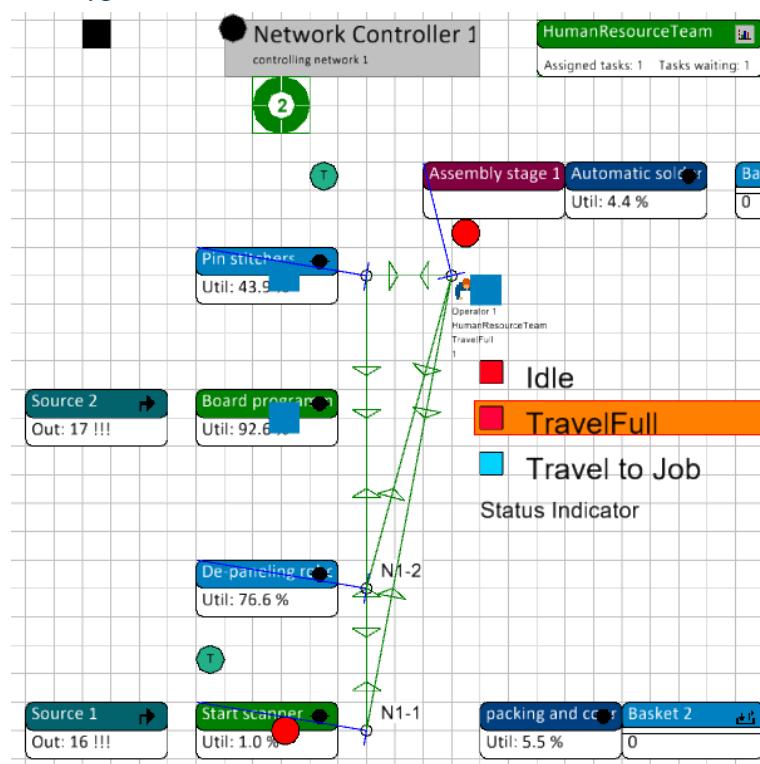


Figure 36 - 1st Prototype

At first, I tried to make the system dynamic in a way that the operator would be the one showing the transfer times and machine operations. This seemed to be harder than I initially expected.

The main hurdles I was trying to solve were the problem of operators deadlocking. And how to prioritize or divide human resource tasks in a way that they would be done in the steps described in "Table 9 - Observation dataset".

Deadlocking happens when the operator is transferring a product node without being able to unload the product node. When this happens, the operator puts itself in a deadlock even when the unload possibility opens afterwards.

To solve this problem, you could check the destination atom if its free before handing of the product. This would take some channel manipulation and consideration, and I concluded I didn't have the time to develop this.

Prioritization or step tasks where difficult to solve. I hypothesized potential risks with prioritization of tasks. This would also have potential risks of deadlocking with the transfers to assembly stage 1. Channel manipulation would be necessary.

I researched within the documentation of Enterprise Dynamics, and I found no possible way to create an order for human resource steps that the operator would be able to follow.

### Layout

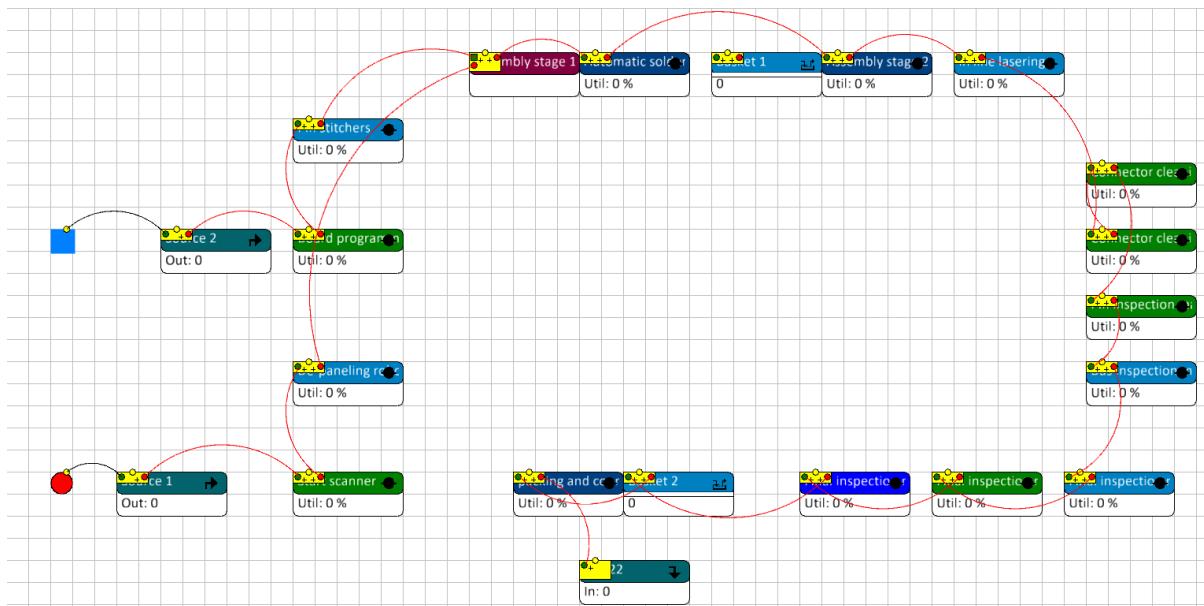


Figure 37 - 2nd Prototype definitive for continuous production

My final design is significantly easier to understand. I decided to change the transfers to just the transfers between atoms. This would mean that the transfer times between atoms need to be simulated within the atoms themselves. There are some drawbacks to this, because it means a less accurate simulation. But no deadlocks would happen.

## Assignment 2

Now that the base is created more information can be added. The first piece of information will be the service time of each machine. In 'Table 9 - Observation dataset' is a time table which contains the service times of each step an operator could take within the production line.



Figure 38 - Server time settings

I implemented the timing table from “Table 9 - Observation dataset” in a way that some servers would also account for the transfer time. There are some problems with this, but this was the only simple option I had to simulate the system without the possibility of having deadlocks.

## Assignment 3

Change the model so that it is able to continuously produce products. Aim for a minimum of 10 products.

### Layout

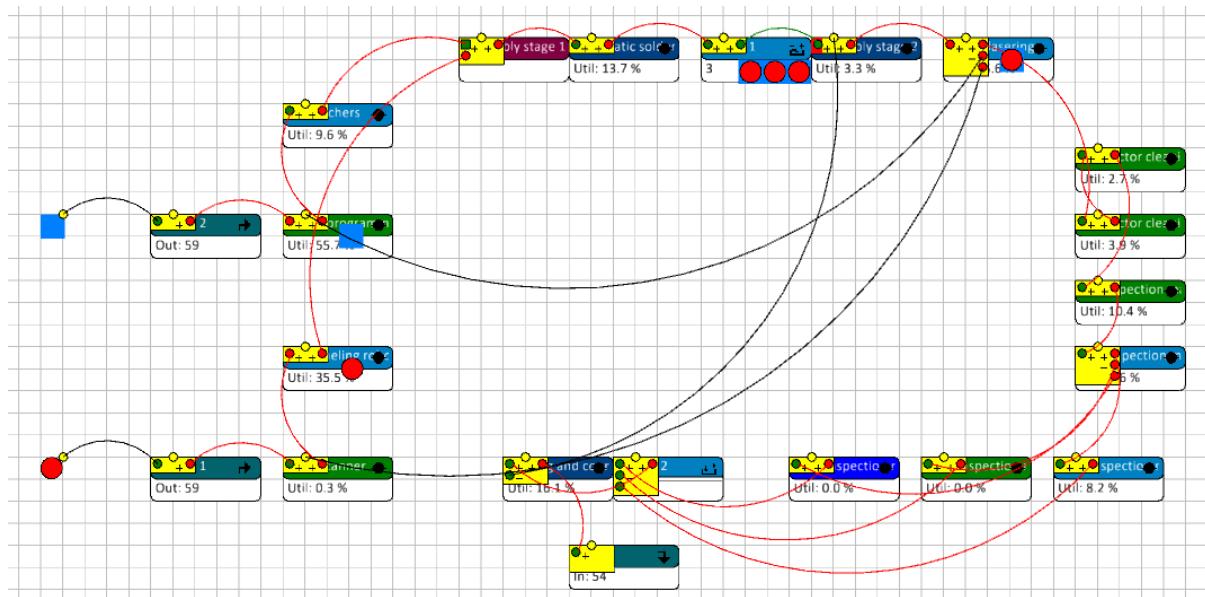


Figure 39 - Continuous production line

No new items are handled until the operator is the end of their line. When that's finished, new product assembly can be started. This way the system can be continuous without any problems.

There seems to be a bottleneck in the basket before assembly stage 2. I hypothesize that this is because the assembly line from operator 1 takes less time compared to the one from operator 2.

## Assignment 4

Create a copy of the previous model. Search the dataset to find equivalent data to the data in the table and change the current parameters to those in the dataset. The original can be used to benchmark the results of the new model. Find the differences between the output and find the bottleneck within the current production line.

//

## Assignment 5

If not done already, use the distribution from the dataset to simulate how a real production line functions. Again, find the differences between the output and find the bottleneck within the current production line. Do this for (at least) 5 runs.

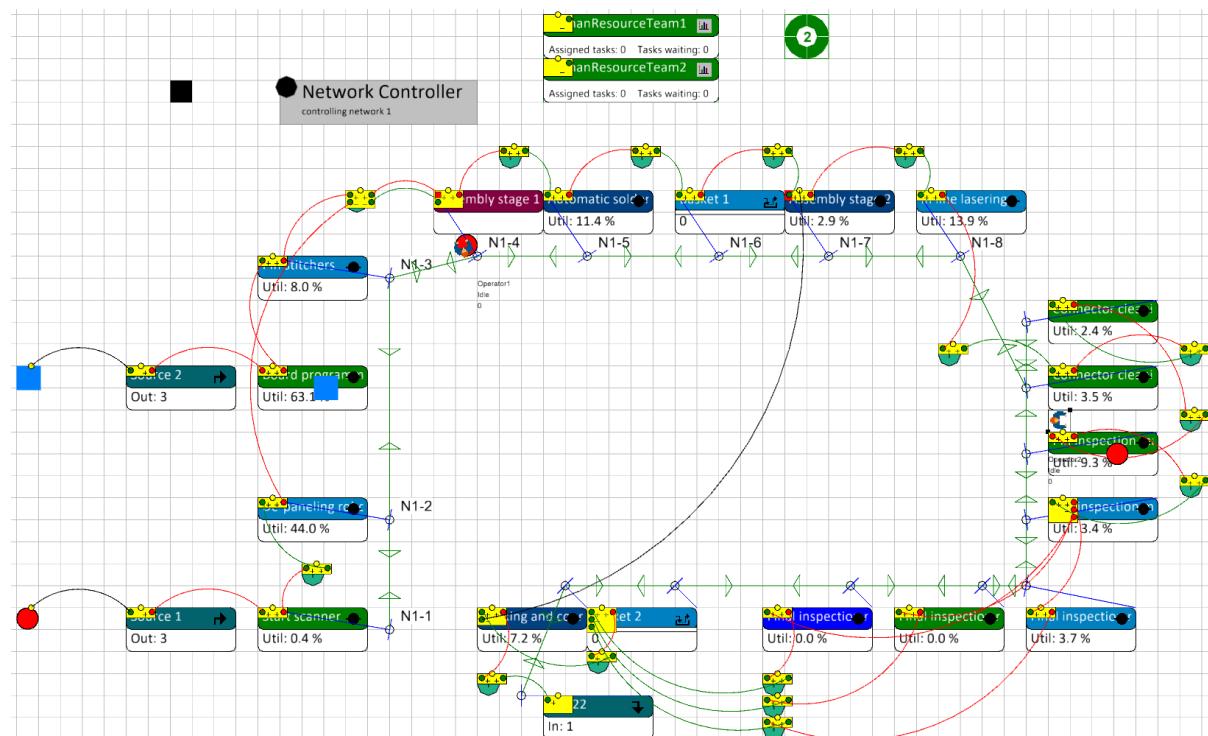


Figure 40 - 3rd Prototype

In “Figure 40 - 3rd Prototype” I decided to change the transfers of products in “Figure 37 - 2nd Prototype definitive for continuous production” to human resource transfers. This would make a more realistic simulation.

To avoid deadlocking I set the speed that the operators travel to a very high level. Unfortunately, I noticed when testing that even with the very high travel speed of the operators the system would deadlock around 1 hour and 47 minutes.

This system can continuously produce at least 9 products before deadlocking.

## Research

### Stochastics (random probability distribution)<sup>1</sup>

Stochasticity refers to the modeling approach of the phenomenon of randomness. But these terms are often used synonymously. Stochastics are often used in probability theory.

Enterprise Dynamics has several built-in stochastic math functions to simulate randomness.

#### Discrete distribution

A discrete distribution applies to a discrete random variable, which can take on a countable set of distinct values. These values can be whole numbers or other countable data points, such as the number of heads in coin tosses, or the number of students in a class.

#### *Poisson distribution*<sup>2</sup>

The Poisson process is a stochastic process with several definitions and applications. It's a counting process, which is a stochastic process in which a random number of points or occurrences are displayed over time. A time-dependent Poisson random variable is defined as the number of points in a process that falls between zero and a certain time. Non-negative numbers make up the index set of this process, but natural numbers make up the state space. Because it can be conceived of as a counting operation, this procedure is often referred to as the Poisson counting process.

The poisson process is a non-negative discrete distribution

#### Continuous distribution

A continuous distribution applies to a continuous random variable, which can take on an infinite number of values within a given range. These values are not countable because they include every possible number between any two points.

#### *Negative Exponential distribution*

## Empirical Distribution

An empirical distribution refers to a probability distribution that is based directly on observed data rather than on a theoretical model. It is used to describe the distribution of a set of observed values, showing the relative frequencies of outcomes in the data.

---

<sup>1</sup> InControl Enterprise Dynamics [Computer program]. (2023). *HTML Help 4DScript category stochastic*. InControl Enterprise Dynamics.

Bartholomew, D. J. (2015). *Stochastic models*. In Elsevier eBooks (pp. 467–472). <https://doi.org/10.1016/b978-0-08-097086-8.43095-3>

Young, J. (2024, May 31). *Discrete Probability Distribution: Overview and examples*. Investopedia. <https://www.investopedia.com/terms/d/discrete-distribution.asp>

OpenAI. (2024). *ChatGPT* (Sep 19 version) [Large language model]. <https://chat.openai.com/chat>

<sup>2</sup> Lendave, V. (2024, July 15). *A guide to stochastic Process and its applications in machine learning*. AIM. <https://analyticsindiamag.com/developers-corner/a-guide-to-stochastic-process-and-its-applications-in-machine-learning/>

## Queueing Theory

### M/M/c queueing model<sup>3</sup>

M/M/C is a multi server system with customers arrival follows a poisson process and exponential service time. When a customer enters an Empty system, he gets the service at once. If the system is non-empty the incoming customer joins the Queue. M/M/C model is a poisson birth death process.

Birth Occurs → Customer arrives

Death Occurs → Customer departs

Both are modeled as Memoryless Markov process.

In M/M/C , M refers to this memoryless/Morkov feature of the arrival and service

## Experiments<sup>4</sup>

### Performance Measures (pfm)

#### *Groups*

The average stay for group experiments is calculated on a weighted basis of total output. This is done to remove biases of different elements in the group with a lower output.

**Average stay:** Weighted average of the elements of the group, divided by sum of the output of the elements k of the group.

$$\frac{\sum_k \text{avgstay}(k) * \text{output}(k)}{\sum_k \text{output}(k)}$$

---

<sup>3</sup> Sundari, S. M., & Srinivasan, S. (2011). *M/M/C queueing model for waiting time of customers in bank sectors*. ResearchGate.

[https://www.researchgate.net/publication/267176376\\_MMC\\_queueing\\_model\\_for\\_waiting\\_time\\_of\\_customers\\_in\\_bank\\_sectors](https://www.researchgate.net/publication/267176376_MMC_queueing_model_for_waiting_time_of_customers_in_bank_sectors)

<sup>4</sup> InControl Simulation Solutions. (2023, November 14). *Experimentation: Enterprise Dynamics 10 (Version 10)*. InControl Simulation Solutions. <https://www.incontrolsim.com>