Assignment 1 (40%)

Simulation for Business Decision Making

Alexander J. Ohrt
UPC
alexander.johansen.ohrt@estudiantat.upc.edu

1 Abstract.

A supermarket is simulated in order to analyze how well it is functioning in some respects. The objective is to quantify how well the industrial system is functioning, by analyzing how many clients can be served per day and how long each client has to wait to pay for their groceries. Some discussion regarding the number of cashiers working is done, in order to give some insight into how the system can be optimized. The conclusion is that this model can be used by managers to quantify how well a given system will work in practice and to optimize the system for profit and customer satisfaction, when coupled with information about expenses.

2 System description, introduction.

2.1 The supermarket (Industrial system)

The supermarket is divided in two different parts: the supermarket and the cashier's area. It is big enough so that there are no problems of delay in the time spent choosing groceries due to the number of people in the supermarket. Therefore, the time needed by the client to choose what he wants to buy only depends on how much items he wants. Of course the more items they choose to buy the more time they will need. Once this action is done the client pays for his groceries.
There are 3 cashiers and one automatic payment station working during the opening hours of the supermarket. There is one single queue for the three cashiers and the station. Each time a cashier (or the automatic station) is available the first client in the line is asked to go to this cashier. The client can decide to avoid to use the automatic cashier station, in that case the first other client waiting in the queue, that agrees to use it, can go for it.
The resulting waiting line does not have any limit. Nevertheless, the shorter it is, the better it is for the client and so for the supermarket.
We want to analyze how many clients by day can be served in this supermarket when three cashiers are working beside the automatic payment station and how long the clients will wait to buy their groceries. The time the clients spend choosing their groceries is not of importance here.

2.1.1 Data

The supermarket is open for 10 hours in a given day, from Monday to Saturday.
Clients arrive in the supermarket following a uniform distribution of 1 to 3 minutes at any period.
The process of choosing the items requires between 4 and 10 minutes for each client independently of the number of clients already being in the same process.
When paying at a cashier the client needs between 3 to 7 minutes to pay (uniform distribution).
When paying at the automatic station the client needs 6 minutes on mean to pay (exponential distribution).
Once the payment is done the client leaves the supermarket following a uniform distribution from 0.5 to 2 minutes.

The clients that come to the supermarket and leave without buying anything are not taken into account.
The probability for a client to avoid to use the automatic cashier station is about 25%.

3 Problem description.

3.1 Hypotheses/Restrictions in the model

Note that the assumptions written in red are not yet validated, but will be used on the model execution.

3.1.1 Structural Hypotheses, not dependent on the implementation.

STR_01: The supermarket owns 3 cashiers. This number is changed at some points in the discussion.
STR_02: The supermarket has one automatic payment station. This number is changed at some points in the discussion.
STR_03. The supermarket should be empty when it closes.

3.1.2 Simplifying Hypotheses, restrictions of the model.

SIM_01: The day of the week (or time of year) will not be considered. Thus the distributions for the arrivals, process of choosing items, paying and leaving are the same for all six days of the workweek, during the entire year.
SIM_02: Lunch break for the cashiers will not be considered.
SIM_03: Change of shifts will not be considered.
SIM_04: We assume that the payment stations don't break during the opening hours.
SIM_05: The people that arrive and leave the store without buying anything are not taken into account.
SIM_06: The waiting line has no limiting length. We will therefore not keep track of the length of the line during the simulation (except for the queue reports that can be found via GPSS).
SIM_07: We do not care about the time spent in the main part of the supermarket, i.e. choosing items from the shelves. This is not part of the objective of the analysis. Time spent shopping only depends on the number of items each customer wants to buy. There is no delay according to how many customers there are at each shelf.

3.1.3 Data Hypotheses

DAT_01: The customer arrival to the supermarket follows a uniform distribution from 1 to 3 minutes.
DAT_02: The process of choosing items follows a uniform distribution from 4 to 10 minutes, independently of the number of customers already in the supermarket.
DAT_03: The time of settling the payment at the manual cashiers follows a uniform distribution from 3 to 7 minutes.
DAT_04: The time of settling the payment at the automatic cashier follows an exponential distribution with mean 6 minutes.
DAT_05: The time a customer spends when leaving the supermarket follows a uniform distribution from 0.5 to 2 minutes.
DAT_06: The probability that a customer wishes to avoid the automatic cashier is 0.25.

DAT_07: The probability given in DAT_06 is independent for each entity in the implementation. This means that each client is free to change their mind when in the queue.

4 Model specification.

In this section, Petri Nets and SDL are used to specify the model.

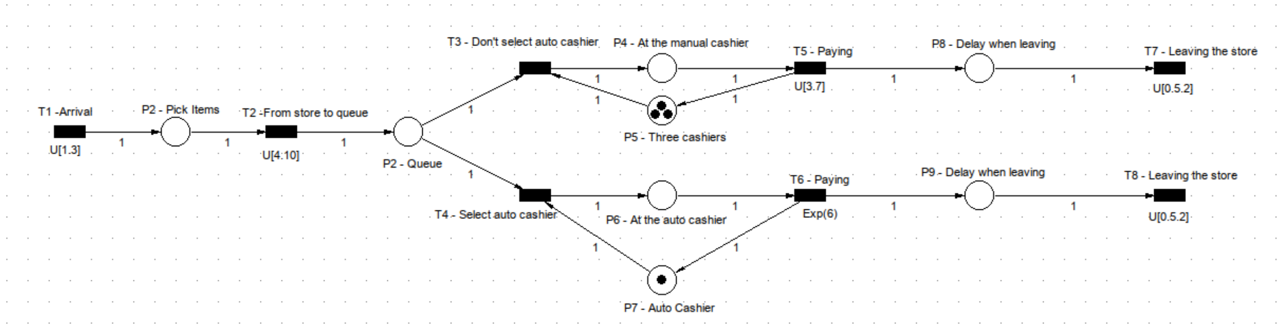

Figure 1: Regular Petri Net

The Petri Net above specifies the model. Each transition is labeled with the relevant distribution it takes to complete, for completeness and simplicity in the modeling. Customers are continuously generated throughout the opening hours, with arrival as described in DAT_01. Notice that we easily can highlight were the different assumptions from section 3 can be seen in the Petri Net. For example, in place P2, we can see that SIM_07 is used, since the transition to the queue does not depend on any other system resources. Moreover, arrival T1 has highlighted a delay according to a U[1,3] as specified in DAT_01. The only assumptions that are not clearly represented in the Petri Net are DAT_06 and DAT_07. This probability decides (to some degree) whether transition T3 or T4 is used when leaving from P2. According to the system assumptions, frequentistically speaking, transition T4 should be used in 75% of the cases where the automatic cashier is available, while transition T3 should be used in the remaining cases (when at least one manual cashier is available and/or when the client refuses to use the available automatic cashier. This refusal takes place in 25% of the cases where the automatic cashier is available). Notice that this Petri Net specification does not take any of this into account explicitly, which is the main drawback of this model specification.

Below, the model is specified using a colored Petri Net, which simplifies the image a bit. Defining the following quantities
- Piece color = {1,2} (1 are customers that choose to use the automatic cashier, 2 are customers that decline using it)
- Machine color = {0,1} (0 are the manual cashiers are 1 is the automatic cashier)
- Var p: machine
- Var p0: piece

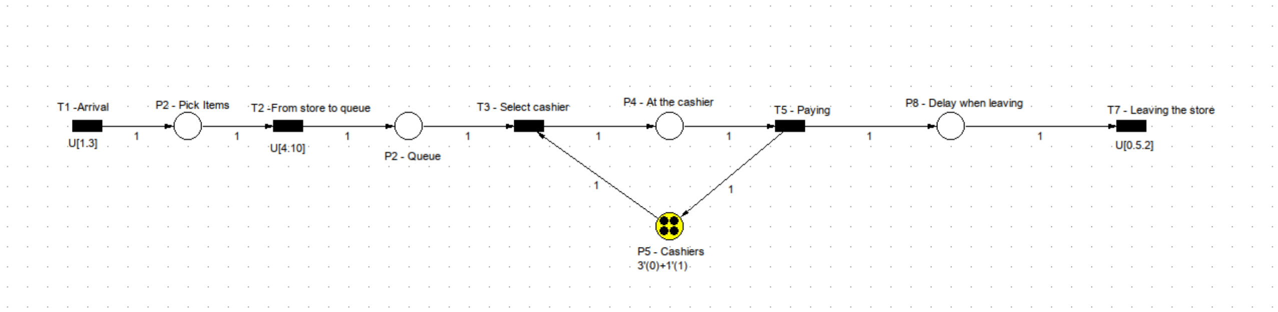

Figure 2: Colored Petri Net

The amount of "p0" has not been specified to begin with, since we do not know this number exactly. We only know that we will simulate the supermarket for 10 hours each labor day.

Next the model is specified using Specification and Description Language (SDL).

First we show the system.

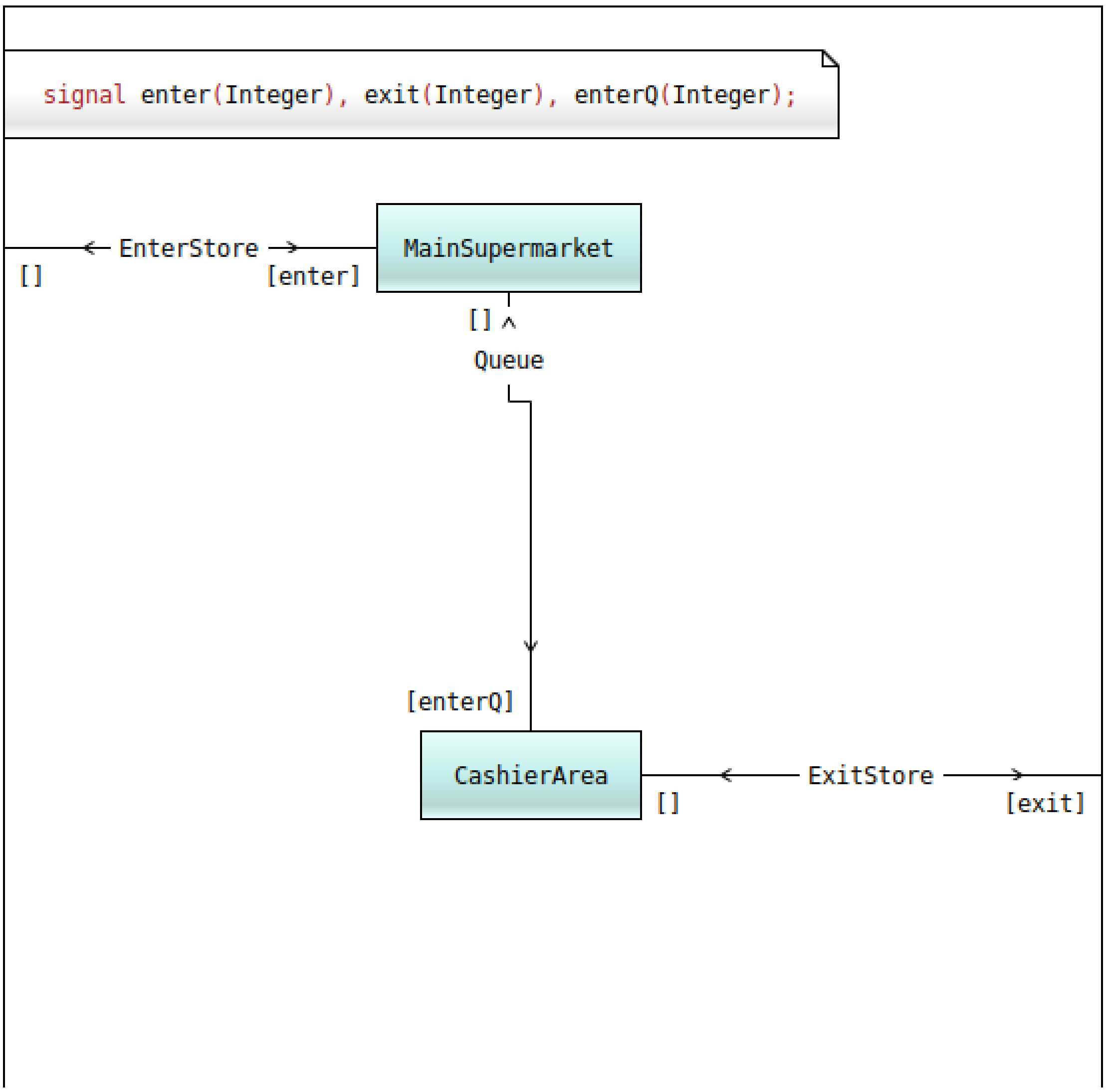

Figure 3: System

Notice that customers cannot leave the store unless they go through the CashierArea. That is, we do not take into account people that go in to the store without buying anything before leaving, i.e. we assume that all customers that enter the store will buy something, as specified in SIM_05. This simplifies our model, but also limits the use of it, without making the model more complex.

Next we show each of the blocks in the system.

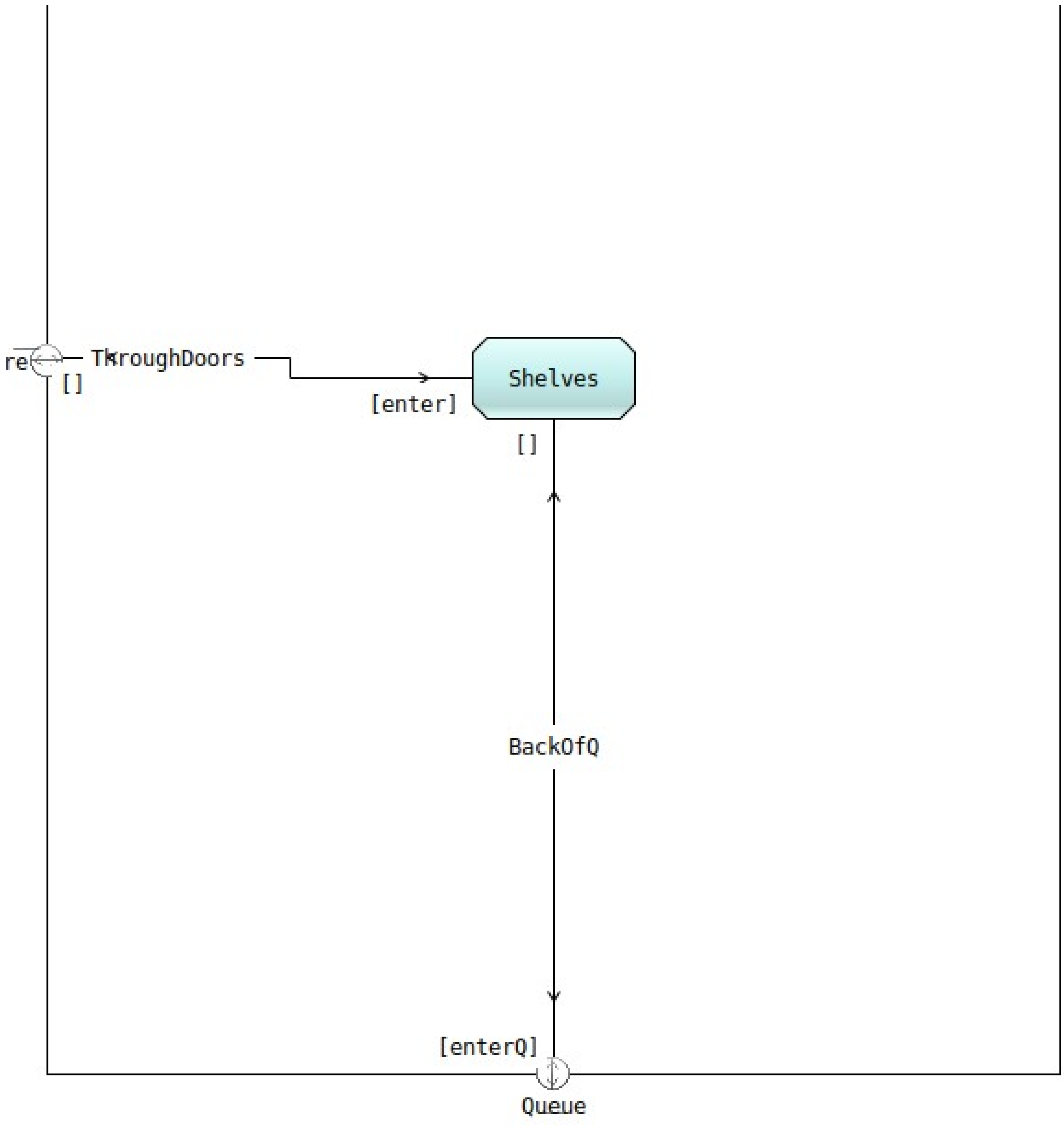

Figure 4: Block: MainSupermarket

The store is entered through the doors (“ThroughDoors”) with the “enter” signal from “EnterStore” (left side of figure 4). Then, from the process “Shelves”, the back of the queue is entered.

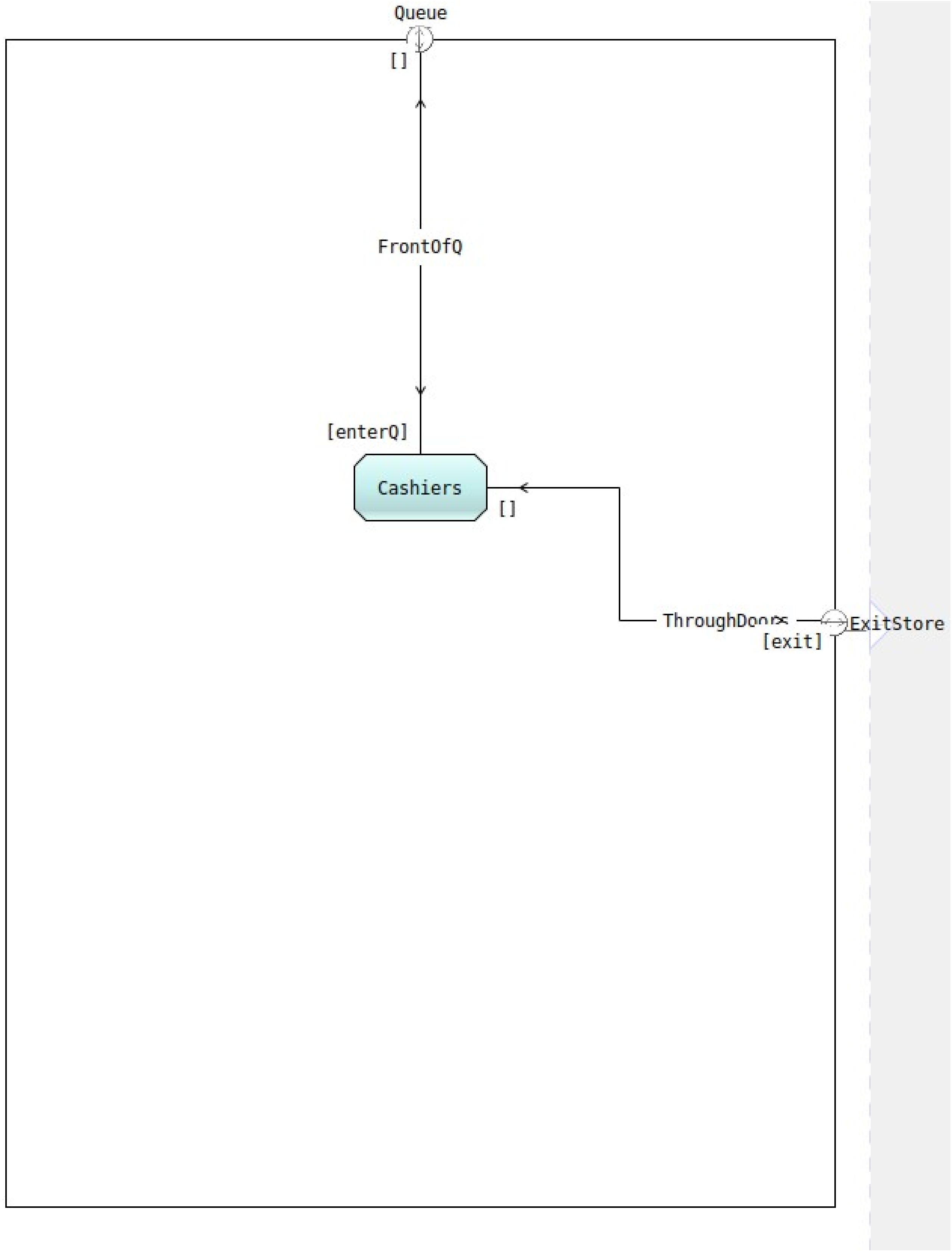

Figure 5: Block: CashierArea

The “Cashiers” process is entered through the front of the queue (“FrontOfQ”), which happens when the signal “enterQ” is sent. After the process of payment at the cashiers is done, the CashierArea is exited through the doors (“ThroughDoors”), when the signal “exit” is sent.

Finally we show the processes in each of the blocks.

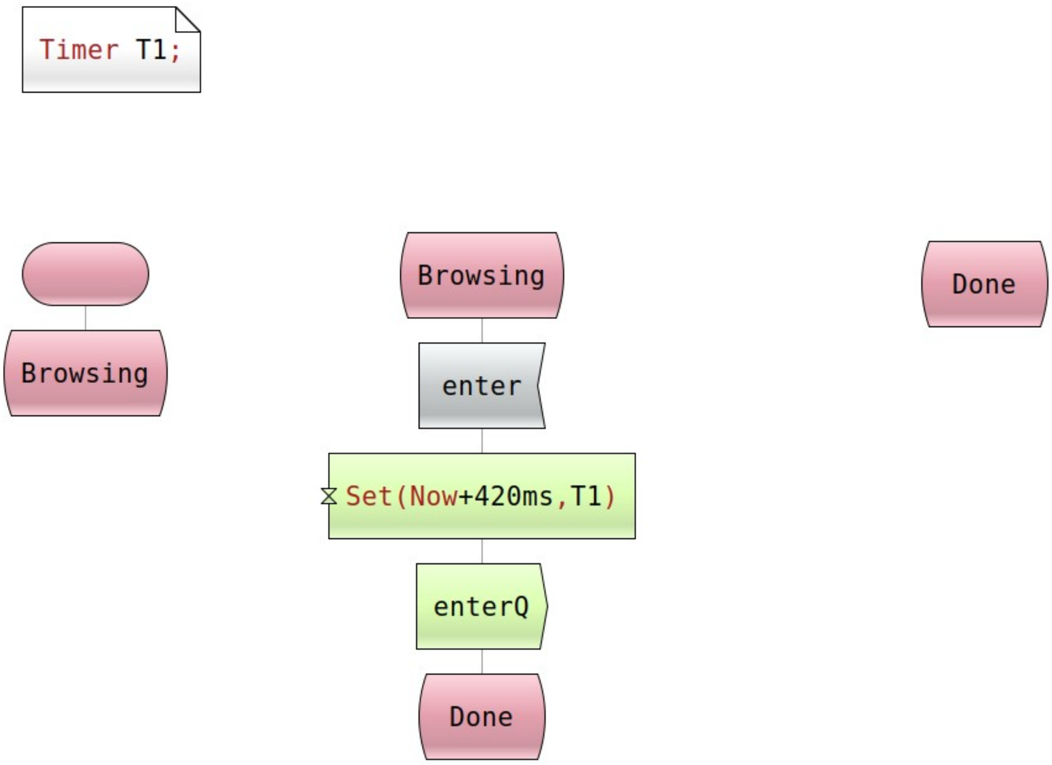

Figure 6: Process: Shelves

Notice that the timer that is shown after the “enter”-signal is a simple timer of 420ms. In reality, according to the assumptions we have made, this timer should follow a U[4,10] (DAT_02). The reason to why this simple timer is used is that I could not find any information in the documentation of SDL on how to add a timer based on a distribution. The time is set to 420 units, because this is the mean value of 4 and 10 minutes (DAT_02), when using seconds as units (7 x 60 seconds is 420 seconds). The reason I have ms instead of s as units is in order to execute the simulation faster. Thus, these choices are a bit arbitrary, but where done in order to execute the SDL model in PragmaDEV. Obviously, this simple timer does not meet the requirements that have been set for the model earlier, and should be fixed before production.

Once the customer is done browsing (i.e. the timer has finished), the “enterQ” signal is sent and the customer reaches the final state “Done” in the “Shelves”-process.

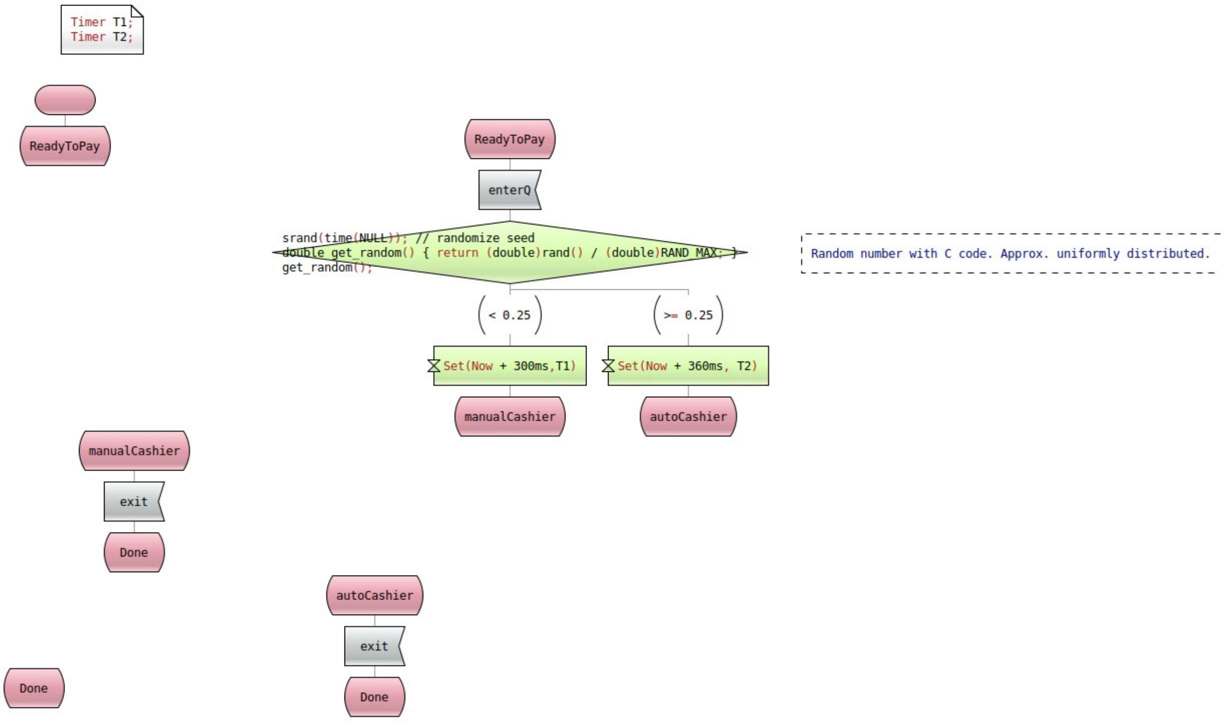

Figure 7: Process: Cashiers

Similar remarks about the timers can be made in the case of the “Cashiers”-process; notice that we have two different simple timers in this case. According to the assumptions mentioned earlier, timer T1 should have followed a U[3,7] (DAT_03), while the timer T2 should have followed an Exponential distribution with mean 6 minutes (DAT_04). I have used 300 ms in timer T1 because the mean of the time used to pay in the manual cashiers is 5 minutes, which translates to 300 seconds. I have changed the units to ms to make the timers run faster during execution. Moreover, I have used 360 ms in timer T2 because of the mean time of 6 minutes to complete the payment, which translates to 360 seconds.

I read in the documentation of SDL that one can use C code inside the “Decision” element; “any standard C expression that returns a C true/false expression” should work. Thus, the C code I have added in the “Decision” element is meant to generate (approximately) uniformly distributed numbers in the interval [0,1]. This is done in order to model the probability of a customer avoiding the automatic cashier, which is assumed to have a probability of 0.25.

Notice that the “Cashiers”-process gives rise to a large simplification of the assumptions I defined earlier. In fact, this is the same error or drawback as is seen in the Petri Net specification earlier. In this process, there has not been implemented any form of test if the manual cashiers or the automatic cashiers are available when the customer wants to pay. In this case, the customer always goes to the given type of cashier after being added to the correct decision branch following the random number generation and after the delay of the timer is complete. This is an error that should not be present, which should be fixed before production. Notice that I have implemented the model in GPSS keeping a special eye on not including the same error in the simulation. The implementation is shown next.

5 Implementation.

The model is implemented using GPSS. The GPSS code is shown below.

```
*Storage for three cashiers.
CASHIERS            STORAGE 3
ARRIVAL             GENERATE 2,1
CHOOSING            ADVANCE 7,3
WAITING_LINE        QUEUE CUA
*
*Availability? If at least one manual cashier is available, enter PAYMENT_CASHIER. If not, check if auto is avail.
AVAILABLE_MAN       GATE SNF CASHIERS,AVAILABLE_AUTO
PAYMENT_CASHIER     ADVANCE 0
                    ENTER CASHIERS
                    DEPART CUA
                    ADVANCE 5,2
                    LEAVE CASHIERS
                    TRANSFER ,LEAVING
*
*Availability? If auto cashier not available, wait here. I do not like this assumption, but back to queue makes loop.
AVAILABLE_AUTO      GATE NU AUTO_CASHIER
*Adding C operator above gives loop; Both with WAITING_LINE and AVAILABLE_MAN.
*25% probability of avoiding automatic. If don't want to use auto, go back to queue.
PAYMENT_METHOD      TRANSFER .25,PAYMENT_AUTO,WAITING_LINE
PAYMENT_AUTO        ADVANCE 0
                    SEIZE AUTO_CASHIER
                    DEPART CUA
                    ADVANCE (EXPONENTIAL(1,6,1))
                    RELEASE AUTO_CASHIER
                    TRANSFER ,LEAVING
*
LEAVING             ADVANCE 1.25,0.75
                    TERMINATE


*Separate transaction simulating the work day (10 hours).
GENERATE 600
TERMINATE 1
```

Figure 8: Implementation in GPSS

Notice that the assumption that no customers should be left in the store when it closes has not been taken into account in the implementation (STR_03). This could be done relatively easily in the code above however and would change the behavior of the system slightly.

Some more discussion around the process of implementing the model in GPSS is given. I have tried to make gates such that

- If there is at least one manual cashier available, go directly to one of these from the queue.
- If there is no manual cashier available, go to the gate in front of the automatic cashier.

This implementation implies that, if the automatic cashier is not available when an entity (here: a client) arrives at the gate in front of the automatic cashier, the entity will wait there until the automatic cashier becomes available. This can be defined as an assumption in the model to begin with, but I think this is too restricting in such a model. Therefore, I will simply ask the reader to notice that this is the case in the implementation, noting that this is a fault in the implementation of the model. When trying to have the client go back in the queue when the automatic cashier is unavailable, this makes a loop in the current implementation, as noted in the implementation comments. Moreover, notice that the implementation has no ability to assure that the same clients that once avoided the automatic cashier always will continue avoiding it after being sent back in the queue. Thus, this follows assumption DAT_07; that each customer can change their mind about avoiding the automatic cashier. In practice, this means that the probability of avoiding the automatic cashier given in DAT_06 is independent for each entity that enters the queue, where it does not matter if the same entity has entered the queue at an earlier point in the simulation.

6 Results and Discussion

The supermarket is simulated during one work day (10 hours, 600 minutes). Some results are shown and discussed in this section.

First we show the results from simulating the supermarket with the same assumptions as given in section 3.

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ARRIVAL	1	GENERATE	305	0	0
CHOOSING	2	ADVANCE	305	3	0
WAITING_LINE	3	QUEUE	321	0	0
AVAILABLE_MAN	4	GATE	321	0	0
PAYMENT_CASHIER	5	ADVANCE	239	0	0
	6	ENTER	239	0	0
	7	DEPART	239	0	0
	8	ADVANCE	239	1	0
	9	LEAVE	238	0	0
	10	TRANSFER	238	0	0
AVAILABLE_AUTO	11	GATE	82	0	0
PAYMENT_METHOD	12	TRANSFER	82	0	0
PAYMENT_AUTO	13	ADVANCE	63	0	0
	14	SEIZE	63	0	0
	15	DEPART	63	0	0
	16	ADVANCE	63	1	0
	17	RELEASE	62	0	0
	18	TRANSFER	62	0	0
LEAVING	19	ADVANCE	300	1	0
	20	TERMINATE	299	0	0
	21	GENERATE	1	0	0
	22	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
AUTO_CASHIER	63	0.711	6.771	1	302	0	0	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
CUA	20	19	321	275	7.295	13.636	95.154	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
CASHIERS	3	2	0	3	239	1	1.987	0.662	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
301	0	600.276	301	19	20		
307	0	601.490	307	0	1		
304	0	602.016	304	2	3		
302	0	602.516	302	16	17		
303	0	604.814	303	8	9		
306	0	605.852	306	2	3		
305	0	606.074	305	2	3		
308	0	1200.000	308	0	21		

Figure 9: Supermarket simulated as specified in the assumptions. Using exactly the same code as in figure 8.

As we can see from the figure above, there are 3 clients left in the main supermarket ("Choosing"). Moreover, there is one client which is in the process of leaving exactly when the workday is over. Finally, there is one client in the process of paying in each of the two types of cashiers when the workday is over.

Among the 305 clients that enter the supermarket during the 10 hours, 300 of them are attended successfully (including the client that is in the process of leaving). This gives a successful attendance rate of approximately 98%. Moreover, we can see that the average time spent in the queue is 13.6 minutes, which amounts to the average time each customer has to wait to be attended by a cashier (or self-attendance at the automatic payment station). Also notice that only 63 customers use the automatic cashier, as opposed to the 239 that use the manual cashiers (counting with the people that are stuck in payment when the workday is over). This amounts to a percentage of approximately 20.9% of attended to clients using the automatic cashier.

In order to contrast the original simulation, the system is simulated with the same specification as before, but with 4 manual cashiers instead of 3.

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ARRIVAL	1	GENERATE	303	0	0
CHOOSING	2	ADVANCE	303	4	0
WAITING_LINE	3	QUEUE	302	0	0
AVAILABLE_MAN	4	GATE	302	0	0
PAYMENT_CASHIER	5	ADVANCE	278	0	0
	6	ENTER	278	0	0
	7	DEPART	278	0	0
	8	ADVANCE	278	2	0
	9	LEAVE	276	0	0
	10	TRANSFER	276	0	0
AVAILABLE_AUTO	11	GATE	24	0	0
PAYMENT_METHOD	12	TRANSFER	24	0	0
PAYMENT_AUTO	13	ADVANCE	21	0	0
	14	SEIZE	21	0	0
	15	DEPART	21	0	0
	16	ADVANCE	21	0	0
	17	RELEASE	21	0	0
	18	TRANSFER	21	0	0
LEAVING	19	ADVANCE	297	1	0
	20	TERMINATE	296	0	0
	21	GENERATE	1	0	0
	22	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
AUTO_CASHIER	21	0.241	6.891	1	0	0	0	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
CUA	5	3	302	294	1.685	3.348	126.370	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
CASHIERS	4	2	0	4	278	1	2.299	0.575	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
299	0	600.721	299	8	9		
298	0	600.813	298	19	20		
301	0	600.884	301	2	3		
300	0	601.310	300	8	9		
305	0	601.337	305	0	1		
304	0	602.992	304	2	3		
302	0	603.445	302	2	3		
303	0	604.054	303	2	3		
306	0	1200.000	306	0	21		

Figure 10: Supermarket simulated with 4 manual cashiers, instead of the original 3.

As we can see from the figure above, there are 4 clients left in the main supermarket ("Choosing"). Moreover, there is one client which is in the process of leaving exactly when the workday is over. Finally, there are two clients in the process of paying at the manual cashiers when the workday is over.

Among the 303 clients that enter the supermarket during the 10 hours, 297 of them are attended successfully. This gives a successful attendance rate of approximately 98%, which is approximately equal to the attendance rate in the system when using 3 manual cashiers. Moreover, we can see that the average time spent in the queue is 3.348 minutes, which is a lot lower than the average time spent with one manual cashier less. Also notice that only 21 customers use the automatic cashier in this case, as opposed to the 278 that use the manual cashiers. This amounts to a percentage of approximately 7% of attended to clients using the automatic cashier. Notice that this percentage is much lower than in the supermarket when having one manual cashier less, rendering the automatic cashier less useful in the system. This makes sense because less clients are attended to by the automatic cashier in this case, while more clients are attended to by the manual cashiers, compared to the first simulated system.

Below, the same system is simulated with 5 manual cashiers. The results from the simulation are shown.

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ARRIVAL	1	GENERATE	291	0	0
CHOOSING	2	ADVANCE	291	3	0
WAITING_LINE	3	QUEUE	288	0	0
AVAILABLE_MAN	4	GATE	288	0	0
PAYMENT_CASHIER	5	ADVANCE	284	0	0
	6	ENTER	284	0	0
	7	DEPART	284	0	0
	8	ADVANCE	284	4	0
	9	LEAVE	280	0	0
	10	TRANSFER	280	0	0
AVAILABLE_AUTO	11	GATE	4	0	0
PAYMENT_METHOD	12	TRANSFER	4	0	0
PAYMENT_AUTO	13	ADVANCE	4	0	0
	14	SEIZE	4	0	0
	15	DEPART	4	0	0
	16	ADVANCE	4	0	0
	17	RELEASE	4	0	0
	18	TRANSFER	4	0	0
LEAVING	19	ADVANCE	284	1	0
	20	TERMINATE	283	0	0
	21	GENERATE	1	0	0
	22	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
AUTO_CASHIER	4	0.047	7.080	1	0	0	0	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
CUA	2	0	288	287	0.009	0.019	5.384	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
CASHIERS	5	1	0	5	284	1	2.330	0.466	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
284	0	601.068	284	19	20		
286	0	601.154	286	8	9		
293	0	601.604	293	0	1		
287	0	601.781	287	8	9		
288	0	602.089	288	8	9		
291	0	602.625	291	2	3		
290	0	603.417	290	2	3		
292	0	604.191	292	2	3		
289	0	605.307	289	8	9		
294	0	1200.000	294	0	21		

Figure 11: Results from simulating a supermarket with 5 manual cashiers, instead of the original 3.

As we can see from the figure above, there are 3 clients left in the main part of the supermarket ("Choosing"). Moreover, there is one client which is in the process of leaving exactly when the workday is over. Finally, there are four clients in the process of paying at the manual cashiers when the workday is over.

Among the 291 clients that enter the supermarket during the 10 hours, 284 of them are attended to successfully. This gives a successful attendance rate of approximately 97.5%, which is approximately equal to the attendance rate in the systems in the earlier simulations. Moreover, we can see that the average time spent in the queue is 0.019 minutes, which is a lot lower than the average time spent with one or two manual cashiers less. This is a clear improvement if queue time is important. Also notice that only 4 customers use the automatic cashier in this case, as opposed to the 284 that use the manual cashiers. This amounts to a percentage of approximately 1.39% of attended to clients using the automatic cashier. Notice that this percentage is much lower than in the supermarket when having one or two manual cashiers less, rendering the automatic cashier less useful in the system.

Some remarks regarding the evolution of the system as the number of manual cashiers is increased: Notice that the automatic cashier is rendered more and more useless, and more and more clients use the manual cashiers when the number of manual cashiers increases. This makes sense since there more often will be a manual cashier available when there are more of them. Moreover, notice that the amount of people still in the process of paying at the manual cashiers at the end of the workday increases with the increase in manual cashiers. This is simply a consequence of the fact that more people are using the manual cashiers closer and closer to the closing time. In practice, by simply allowing the people to complete their payments, this would lead to the store getting locked a couple of minutes after the real closing time.

Without showing the simulation results, adding 6 manual cashiers gives a percentage of approximately 0.68% of attended to clients using the automatic cashier, while 7 manual cashiers will render the automatic cashier completely useless, with 0 clients using it.

We could also simulate the model with fewer manual cashiers. Below, the results from simulating with 2 cashiers is shown.

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ARRIVAL	1	GENERATE	309	0	0
CHOOSING	2	ADVANCE	309	3	0
WAITING_LINE	3	QUEUE	323	0	0
AVAILABLE_MAN	4	GATE	323	47	0
PAYMENT_CASHIER	5	ADVANCE	175	0	0
	6	ENTER	175	0	0
	7	DEPART	175	0	0
	8	ADVANCE	175	1	0
	9	LEAVE	174	0	0
	10	TRANSFER	174	0	0
AVAILABLE_AUTO	11	GATE	101	0	0
PAYMENT_METHOD	12	TRANSFER	101	0	0
PAYMENT_AUTO	13	ADVANCE	84	0	0
	14	SEIZE	84	0	0
	15	DEPART	84	0	0
	16	ADVANCE	84	1	0
	17	RELEASE	83	0	0
	18	TRANSFER	83	0	0
LEAVING	19	ADVANCE	257	0	0
	20	TERMINATE	257	0	0
	21	GENERATE	1	0	0
	22	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
AUTO_CASHIER	84	0.980	7.000	1	202	0	0	47	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
CUA	65	64	323	184	33.073	61.437	142.763	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
CASHIERS	2	1	0	2	175	1	1.461	0.731	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
311	0	600.390	311	0	1		
202	0	603.103	202	16	17		
308	0	603.879	308	2	3		
310	0	604.773	310	2	3		
309	0	604.984	309	2	3		
307	0	606.636	307	8	9		
312	0	1200.000	312	0	21		

Figure 12: Results from simulating the supermarket with 2 manual cashiers, instead of the original 3. The same number of automatic cashiers is used.

As we can see from the figure above, there are 3 clients left in the main supermarket ("Choosing"). Moreover, there is one client which is in the process of paying at each of the two types of cashiers respectively exactly when the workday is over. Finally, a number that is more worrying; 47 clients are waiting to be attended to at the gate in front of the manual cashiers. Thus, we have identified a clear bottleneck in this system: since the automatic cashier still is in use when the workday is over, and because there are not enough manual cashiers, these people are still waiting at the gate. Hence, this system does not look to be a good way to organize a supermarket.

Among the 309 clients that enter the supermarket during the 10 hours, 257 of them are attended to successfully. This gives a successful attendance rate of approximately 83%, which is a significant decrease compared to the rates seen earlier. Moreover, we can see that the average time spent in the queue is 61.437 minutes, which is a significant increase compared to earlier supermarket simulations. This is a clear deterioration in average queue time. Notice that 84 customers use the automatic cashier in this case, while 175 clients use the manual cashiers. This amounts to a percentage of approximately 32% of attended to clients using the automatic cashier. Notice that this percentage is much larger than what is seen earlier, which could be seen as a good indication that the automatic cashier is useful. However, the worrying reason behind this is simply that there are too few manual cashiers, which means that people are stuck waiting (can also be because of too few automatic cashiers, but this amount is fixed at this stage in the analysis). All in all, customers entering this supermarket would surely not be very happy with the service they received.

In order to rid the bottleneck found in the last simulation, we try to keep the number of manual cashiers fixed while adding another automatic cashier. In order to follow through with this experiment, the code given in figure 8 has to be changed. A new implementation is given in the figure below.

```
*Storage for three cashiers.
CASHIERS            STORAGE 3
AUTO_CASHIERS       STORAGE 1
ARRIVAL             GENERATE 2,1
CHOOSING            ADVANCE 7,3
WAITING_LINE        QUEUE CUA
*
*Availability? If at least one manual cashier is available, enter PAYMENT_CASHIER. If not, check if auto is avail.
AVAILABLE_MAN       GATE SNF CASHIERS,AVAILABLE_AUTO
PAYMENT_CASHIER     ADVANCE 0
                    ENTER CASHIERS
                    DEPART CUA
                    ADVANCE 5,2
                    LEAVE CASHIERS
                    TRANSFER ,LEAVING
*
*Availability? If auto cashier not available, wait here. I do not like this assumption, but back to queue makes loop.
AVAILABLE_AUTO      GATE SNF AUTO_CASHIERS
*Adding C operator above gives loop; Both with WAITING_LINE and AVAILABLE_MAN.
*25% probability of avoiding automatic. If don't want to use auto, go back to queue.
PAYMENT_METHOD      TRANSFER .25,PAYMENT_AUTO,WAITING_LINE
PAYMENT_AUTO        ADVANCE 0
                    ENTER AUTO_CASHIERS
                    DEPART CUA
                    ADVANCE (EXPONENTIAL(1,6,1))
                    LEAVE AUTO_CASHIERS
                    TRANSFER ,LEAVING
*
LEAVING             ADVANCE 1.25,0.75
                    TERMINATE


*Separate transaction simulating the work day (10 hours).
GENERATE 600
TERMINATE 1
```

Figure 13: New implementation of system with possibility of adding more automatic cashiers. Notice that this exact implementation still only has one automatic cashier, since the AUTO_CASHIERS storage unit is 1.

In order to check that the code is written correctly, we simulate the system with one automatic cashier in the AUTO_CASHIERS storage, to see if the results are the same as in figure 9.

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ARRIVAL	1	GENERATE	305	0	0
CHOOSING	2	ADVANCE	305	3	0
WAITING_LINE	3	QUEUE	321	0	0
AVAILABLE_MAN	4	GATE	321	0	0
PAYMENT_CASHIER	5	ADVANCE	239	0	0
	6	ENTER	239	0	0
	7	DEPART	239	0	0
	8	ADVANCE	239	1	0
	9	LEAVE	238	0	0
	10	TRANSFER	238	0	0
AVAILABLE_AUTO	11	GATE	82	0	0
PAYMENT_METHOD	12	TRANSFER	82	0	0
PAYMENT_AUTO	13	ADVANCE	63	0	0
	14	ENTER	63	0	0
	15	DEPART	63	0	0
	16	ADVANCE	63	1	0
	17	LEAVE	62	0	0
	18	TRANSFER	62	0	0
LEAVING	19	ADVANCE	300	1	0
	20	TERMINATE	299	0	0
	21	GENERATE	1	0	0
	22	TERMINATE	1	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
CUA	20	19	321	275	7.295	13.636	95.154	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
CASHIERS	3	2	0	3	239	1	1.987	0.662	0	0
AUTO_CASHIERS	1	0	0	1	63	1	0.711	0.711	0	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
301	0	600.276	301	19	20		
307	0	601.490	307	0	1		
304	0	602.016	304	2	3		
302	0	602.516	302	16	17		
303	0	604.814	303	8	9		
306	0	605.852	306	2	3		
305	0	606.074	305	2	3		
308	0	1200.000	308	0	21		

Figure 14: Supermarket simulated with possibility of more than one automatic cashier. However, only one storage unit for automatic cashiers is added in this case, in order to check if implementation is comparable to earlier.

As seen in the figure above, indeed the results are the same as in figure 9 and the implementation seems correct.

Now, we add another storage unit for the automatic cashiers, simulating the system with 2 manual cashiers and 2 automatic cashiers. The results from this system are given below.

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
ARRIVAL	1	GENERATE	305	0	0
CHOOSING	2	ADVANCE	305	5	0
WAITING_LINE	3	QUEUE	340	0	0
AVAILABLE_MAN	4	GATE	340	1	0
PAYMENT_CASHIER	5	ADVANCE	173	0	0
	6	ENTER	173	0	0
	7	DEPART	173	0	0
	8	ADVANCE	173	1	0
	9	LEAVE	172	0	0
	10	TRANSFER	172	0	0
AVAILABLE_AUTO	11	GATE	166	0	0
PAYMENT_METHOD	12	TRANSFER	166	0	0
PAYMENT_AUTO	13	ADVANCE	126	0	0
	14	ENTER	126	0	0
	15	DEPART	126	0	0
	16	ADVANCE	126	2	0
	17	LEAVE	124	0	0
	18	TRANSFER	124	0	0
LEAVING	19	ADVANCE	296	0	0
	20	TERMINATE	296	0	0
	21	GENERATE	1	0	0
	22	TERMINATE	1	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
CUA	42	41	340	254	17.112	30.198	119.387	0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
CASHIERS	2	1	0	2	173	1	1.436	0.718	0	0
AUTO_CASHIERS	2	0	0	2	126	1	1.486	0.743	1	0

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
302	0	600.468	302	2	3		
298	0	600.493	298	16	17		
307	0	600.988	307	0	1		
297	0	601.862	297	16	17		
303	0	602.052	303	2	3		
304	0	602.246	304	2	3		
301	0	602.356	301	8	9		
305	0	603.491	305	2	3		
306	0	607.885	306	2	3		
308	0	1200.000	308	0	21		

Figure 15: Supermarket simulated with two manual cashiers and two automatic payment stations.

As we can see from the figure above, there are 5 clients left in the main supermarket ("Choosing"). Moreover, there is one client in the process of paying at a manual cashier and two clients in the process of paying at the automatic payment station. Notice that only one client is stuck at the gate in front of the manual cashiers at the end of the labor day. Remember that we found a bottleneck in figure 12, when employing 1 automatic payment station and 2 manual cashiers. From this result, we can already see that the addition of another automatic payment station to this configuration has gotten rid of the bottleneck.

Among the 305 clients that enter the supermarket during the 10 hours, 300 of them are attended to successfully. This gives a successful attendance rate of approximately 98%. Notice that these statistics are exactly the same as in the first simulation with 3 manual cashiers and one automatic station, as seen in figure 9 and 14. However, we can see that the average time spent in the queue is 30.198 minutes, which is a significant increase compared to the results shown in figure 9 and 14. This is a clear deterioration in average queue time. Notice that 126 customers use the automatic cashier in this case, while 173 clients use the manual cashiers. This amounts to a percentage of approximately 42% of the attended to clients using the automatic cashiers.

Notice that there are many more different variations of the system that can be simulated. One could argue that the distributions of entering, leaving, choosing items and paying at each of the types of cashiers should be differently defined. Moreover, disregarding the definitions of the distributions, one could try many more configurations of manual cashiers and automatic cashiers, in order to quantify what sort of distribution of types of cashiers would be better for the waiting time in the queue (reducing it as much as possible). These types of simulations can be very valuable for a manager, since they can be used to make conclusions regarding the added value of investing in more manual cashiers and more employees or more automatic cashiers, taking into account the cost of each of these investments, within the limitations of the defined simulation model.

7 Conclusions

We have seen that there are advantages with adding more manual cashiers and automatic cashiers (obviously); the average time spent in the queue is reduced. Moreover, some configurations of the two types of cashiers are clearly better than others when it comes to the number of clients that can be served per day and for how long each client has to wait in the queue (on average). The original configuration of the supermarket with 3 manual cashiers and one automatic payment station seems to be a reasonably well-working middle ground between the models with too few cashiers, which would be cheaper for the manager, but perform badly when it comes to customer service, and the models with several more cashiers, which perform much better when it comes to customer service, but obviously are more costly to run. Without having any information about the expenses of each of the elements in the system, especially the salary of a cashier, as well as the cost of installing and running the payment stations, we cannot conclude on which configuration would be the best for a given situation. The conclusion is that the model presented and implemented in this work can be used by a manager for two objectives; it can be used to describe how a specific configuration of supermarket would perform in practice and it can be used to optimize the system for maximum customer satisfaction and profit, when coupled with information about the expenses.