UNIVERSITY OF PISA

Performance Evaluation Project

Christmas shop

Professors:
G. Stea, A. Virdis

Authors:
Chiara Caiazza, Gionatan Gallo, Lorenzo Biagini
18/01/2016

During Christmas period, the tills of a large shop are required to checkout and gift-wrap the purchased goods. The analysis indicate which policy is better between united-tills (both check out and gift-wrap are performed in one till) and separated-tills (gift-wrap is performed in separated tills).

Specifications

During Christmas period, the tills of a large shop are required to check out and gift-wrap (with a probability ***p***) the purchased goods. The total number of tills is ***N***. Two policies can be selected:

- Both check out and gift-wrap are performed in one till.
- Gift-wrap is performed in separated tills. The number of tills for this purpose is ***K***.

Customer inter-arrival times, check out times and gift-wrap times are IID RV and represent the system workload. The queuing and response time are the parameters to be studied in both the policies under a varying workload.

Assumptions

Each customer only buys one good. The only considered times are the inter-arrival times of the customers and the time they spend:

- in the queue for checking out
- for the checkout
- in the queue for gift-wrapping
- for the gift-wrap

E.g. the time needed by the customer to go from a checkout till to a gift-wrap till is not taken into account.

The time expressed in seconds for the checkout can be U(25, 45) or exp(35), while for the gift-wrap it can be U(60, 120) or exp(90). During the simulation, the numbers of checkout and gift-wrap tills do not change.

The simulation's factors are:

- ***policy used*** ∈ {united-tills, separated-tills}
- ***N*** ∈ [2, 10], N ∈ ℕ
- ***K*** ∈ [1, n-1], K ∈ ℕ
- ***customers inter-arrival times*** (to simulate peak hours and non-peak hours)
- ***p*** ∈ {0.3, 0.5, 0.7}

The values for N and K are selected according to the other factors and those which lead to an unstable system (i.e. make the queue grow indefinitely) are discarded.

All confidence intervals presented stand at 99%.

Considerations about response times are analogous to those about queuing time (response time appears translated, with respect to the queuing time, by a value equal to the mean of the service time).

Indices

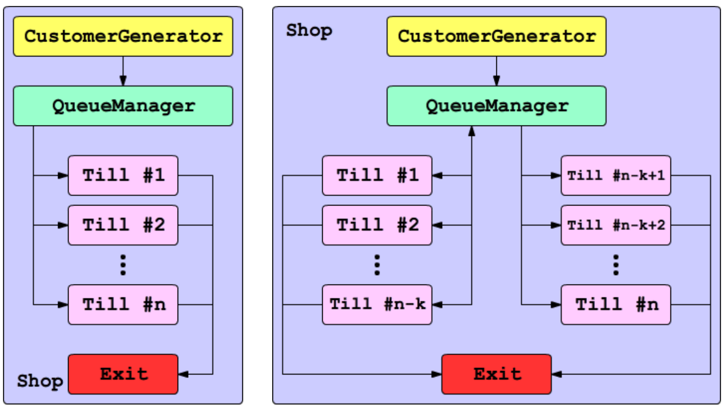
AT = arrival time
LT = leaving time
QT = queuing time
RT = response time
t_x = service time for x
U = till utilization (activity time/total time)
$\bar{y}$ = the mean value of the random variable y
UD = uniform distribution
ED = exponential distribution

Model

The compound module ***Shop*** is composed by 4 types of simple modules:

- ***CustomerGenerator*** generates customers with a random time interval the distribution of which is a simulation factor. It establishes whether the customer needs a gift-wrap and sends him/her to *QueueManager*.
- ***QueueManager*** sends the customer to the till with the shortest queue, also considering if it is serving someone. If there is more than one till serving the same least number of customers, the customer is sent to a random till among these.
- ***Till*** serves the arriving customer when free or enqueues him/her if it is already serving. Once a customer is served, if no other service is needed, he/she is sent to the *Exit* and the customer next in line is immediately served. If the policy selected expects non-separated tills and the customer needs the gift-wrap service, it is immediately performed after checkout. Otherwise, the customer is sent back to *QueueManager* which forwards him/her to the right gift-wrapping till.
- ***Exit*** is used to emit statistics on customers and to delete them

The maps are the following:



United tills ***Separated tills***

Validation

The validation has been performed using fixed values as input to check if the system gives expected output values.

Consistency test: halving the mean of the arrival time of the customers requires N to be doubled. Furthermore, in both policies the following is verified: $\overline{RT} - \overline{QT} \cong \bar{t}_c + \bar{t}_w \cdot p$.

Calibration

Arrival rate has been set in order to have a reasonable number of tills. For peak hours the mean stands at 15 seconds, while for non-peak hours the mean stands at 60 seconds.

In order to have the same utilization of all the tills, the queuing policy provides that the customer shall be assigned to a random till among those which have the same lowest number of customers being served.

The service time for wrapping the good has been set 3.5 times larger than the checkout time in order to point out how the queuing time varies in the two policies when goods are requested to be wrapped. Wrapping time has a mean of 105 s while checkout time has a mean of 30 s.

Warmup time: using a low arrival rate allows the system to reach stability faster, while for the peak hours the steady state is reached more slowly. For this reason the warm up time has been set to 10 ks for all simulations. At this point all simulations show that the mean of the response/queuing time does not vary as before (the curves are almost flat).

Simulation time: in order to have a consistent number of sample at steady state (and in order to be sure to have reached it) the simulation time has been set to 12 h.

Objectives

Find:

- If the two policies require the same number of tills N, other factors being equal
- How the queuing time varies according to the policies
- An explanation to the trend found in the previous bullet
- What changes varying the workload
- Basic relations between factors
- The differences in using the uniform distribution for random variables instead of the exponential distribution
- If separated tills are fairer for those who do not want to wrap their good

Simulation analysis

The assumptions need to be compared with what happens in a real model in order to foresee what is possible to obtain with simulations. Thinking of a queue at the supermarket, there are customers waiting to be served most of the time, because a till with no client can be closed. In this situation of saturation the tills work with an utilization close to 1 and the balance is maintained until the number of customers leaving the shop is equal to the number of customers approaching the tills. If the arrival rate increases the queues get longer and another till is opened, otherwise the tills utilization decreases, a till stops working and can be closed. So the number of open tills can be dynamically changed in order to have them always on work (decrease N) and to avoid too long queues (increase N). By assumptions, however, the number of open tills and the mean of the arrival time do not change during a simulation. Which means that every simulation only describes a scenario out of three (once K, N and arrival rate are set):

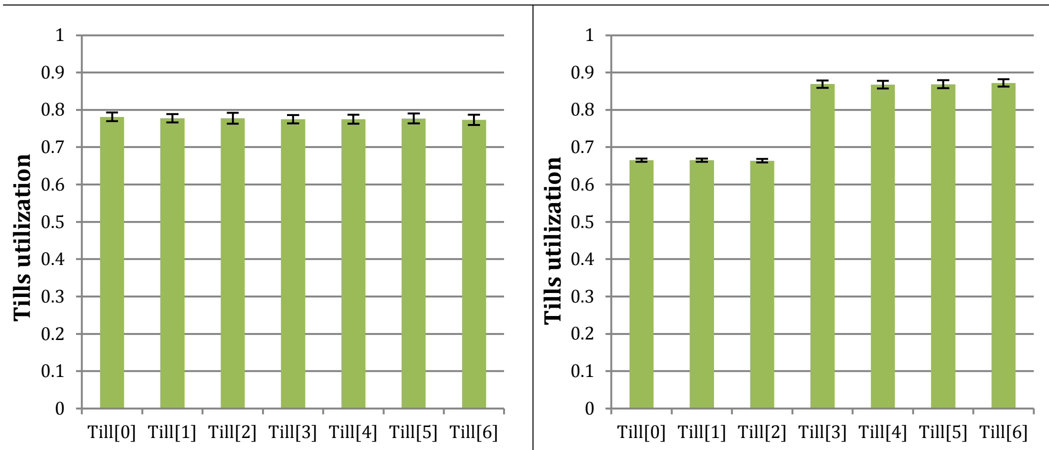
- If $AT < LT$, then the utilization of the tills is less than 1 and there are some periods where tills are idle
- If $AT \simeq LT$, then the utilization of the tills is almost 1 and there is a condition of saturation where the steady state strongly depends on the initial seeds of the random variables

- If $AT > LT$, then the steady state is never reached and the QT diverges

Only the first scenario can be correctly analyzed as the second one has the mean of queuing/response time too dispersed and the third one is not stable: configurations which do not lead to $AT < LT$ are discarded.

Consider now the simplest system where the lowest number of tills can be used: it can be obtained with a high customer arrival time. The policy *united-tills* will require just one till (for both checkout and wrapping), while the policy *separated-tills* will require at least two (one for the checkout and one for the gift-wrapping). So, with an arrival time less or equal to the sum of the services times, the second policy requires one till more than the first one. It has been then analyzed if this condition holds changing the factors, and what happen to queuing/response time. It has been found that *united-tills* reaches the stability with one till less in almost all the simulations (with K in *separated-tills* chosen in order to have the least mean queuing time). So queuing time is shorter in the case of *united-tills*, N being equal. The behavior experienced can be explained by the following observations and charts:

Distribution = *ED* $N = 7$ $K = 4$	$p = 0.5$ $\overline{AT} = 15$ s



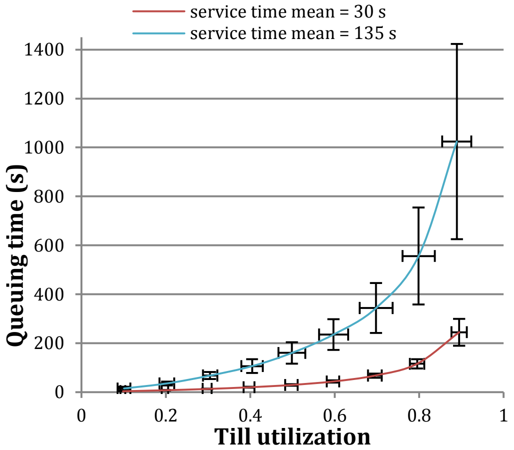
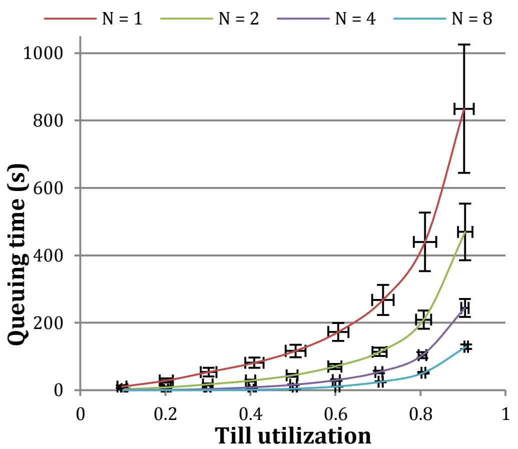
United tills ***Separated tills***

In all the simulations the trend is like the one shown above:

- *United-tills* policy has the same utilization for all the tills

In *separated-tills*, tills performing the same operation have the same utilization (as they are offering different services with different service time mean)

Let $\bar{t}_c$ be the service time mean of a till which only does checkout, $\bar{t}_w$ be the service time mean of a till which only wraps and $\bar{t}_{c+w}$ be the service time mean of a till which performs checkout and wrapping. $\bar{t}_c$ and $\bar{t}_w$ are found in *separated-tills* while $\bar{t}_{c+w}$ is the service time for the *united-tills*. The relation is $\bar{t}_{c+w} = \bar{t}_c + p \cdot \bar{t}_w$. Having set $\bar{t}_c = 30$ s, $\bar{t}_w = 105$ s and $p = \{0.3, 0.5, 0.7\}$, it is $\bar{t}_c < \bar{t}_{c+w} < \bar{t}_w$.

N = 1 *Distribution* = *ED*	*Service time mean* = 105 s *Distribution* = *ED*
	
QT varying service time mean	***QT varying N***

The following statements are possible[1]:

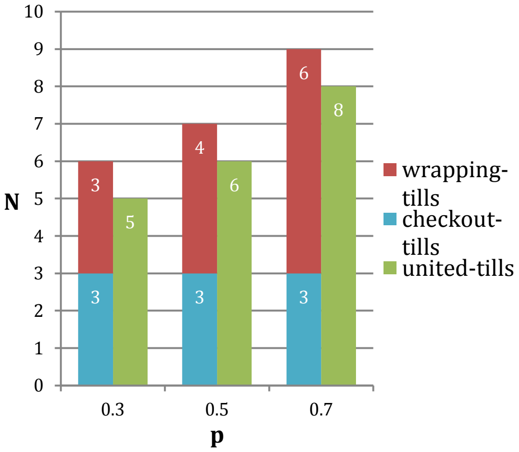
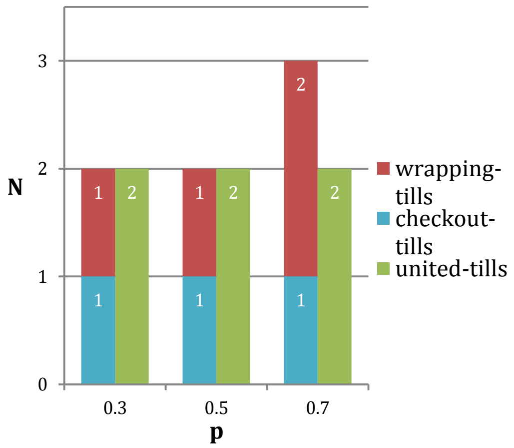
- If the service time mean decreases, *N* and *U* being equal, *QT* does the same (left chart). In other words, the more a service takes, the longer the expected *QT*
- If the number of tills for one operation increases, *QT* decreases, $\bar{t}_x$ and *U* being equal (right chart). This indicates that the more the work is distributed across service units, the shorter *QT*

The last two points plus the relation between $\bar{t}_c$, $\bar{t}_w$ and $\bar{t}_{c+w}$ explain why *united-tills* policy leads, on average, to a smaller queuing time: referring to the simulation shown before

- Customers going in the shop which uses *united-till* policy will experience a longer service time mean ($\bar{t}_{c+w}$) but they will be well distributed across 7 tills (a longer service time mean would make queueing time increase but this is neutralized by distributing the service).
- Customers going in the shop which uses *separated-till* policy may enqueue themselves twice: for the first queuing (checkout only) there is a shorter service time mean ($\bar{t}_c$) but the service is much less distributed (as only 3 tills are used for checkout). Then, those who need the good to be wrapped will stay in queue longer than they would have stayed if the shop had used an *united-tills* policy (as $\bar{t}_w > \bar{t}_{c+w}$ and the service is distributed on 4 tills instead of 7). As a result, this effect makes the mean duration of the two queues to be longer than the one in the other policy. *Separated-tills* can be also worse for those who need to checkout only (when the effect of the bad service distribution overcomes the effect of having a smaller service time mean for the checkout or a smaller *U*), as a consequence, waiting where customers can also ask to wrap their goods is better than waiting where customers can only pay.

The next analysis are meant to show relations between factors. The following dependencies can be assessed:

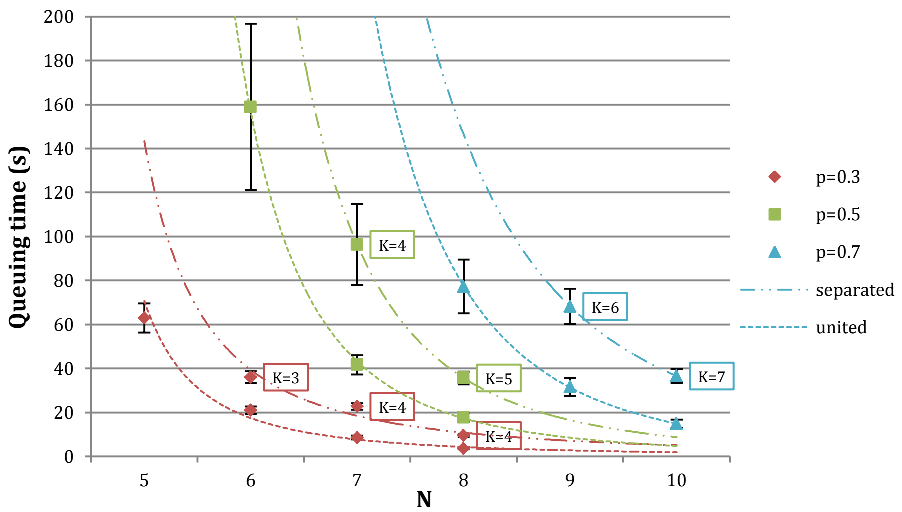
[1] Each curve of the charts has been obtained changing the arrival rate in each simulation, the other factors being equal, in order to have a utilization of the tills close to a specified fraction (*U* = {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9})

$\overline{AT}$ = 15 s *Distribution = ED*	$\overline{AT}$ = 60 s *Distribution = ED*
	

N required in both policies increasing p (with different arrivar time)

N and *K* strongly depend on $\overline{AT}$ and *p*. This is obvious as the larger the amount of customers/requests is, the higher the number of the tills needed to meet the service demand (and both the arrival rate and *p* increase the load of the system). The bar-chart shows the configuration with the lowest number of tills in the two policies in order to have a non-saturated system: the *p* factor influences *N* in both cases but, in the *separated-tills*, only the number of those which wrap changes. As previously indicated, *united-tills* policy reaches the non-saturated state with less tills. The chart on the right shows that as the arrival time decreases, the number of tills needed for the system to be stable decreases as well.

$\overline{AT}$ = 15 s
Distribution = ED

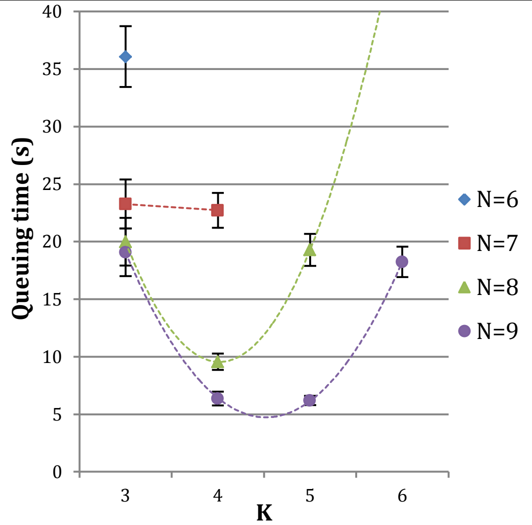


How QT varies in both policies using different N

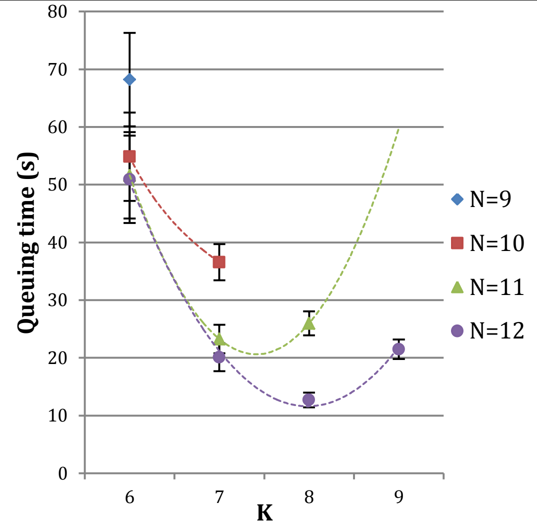
The chart shows the performance of the two policies (the curves with labels are referring to *separated-tills* policy)[2]. It is possible to notice how the regression curves for the queuing time of *separated-tills* are always above the *united-tills* curves.

[2] Also in these simulations *N* has been chosen in order to have a non-saturated system. The presented *K*s are the ones which make the $\overline{QT}$ the minimum

$\overline{AT}$ = 15 s — *Policy* = separated-tills
Distribution = *ED*



Relation between QT and K (p = 0.3)



Relation between QT and K (p = 0.7)

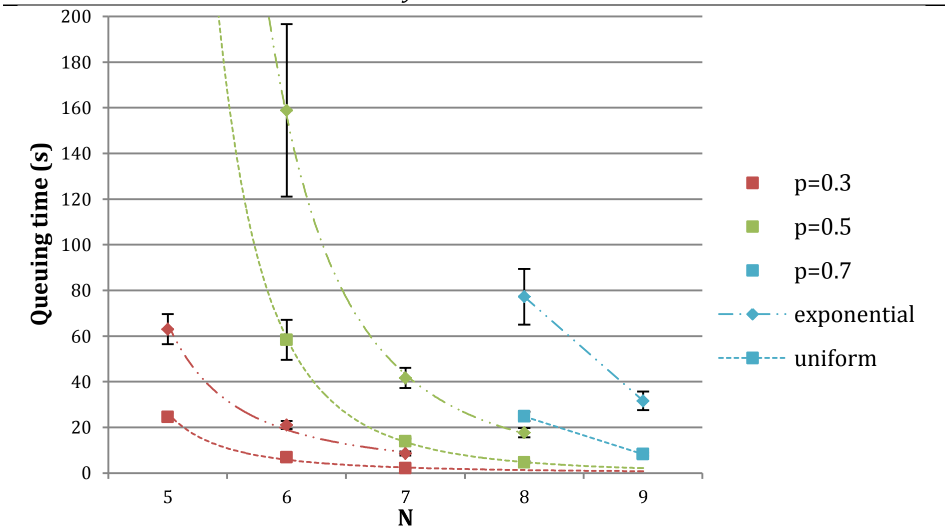
Once the steady state is reached with $N = n$, at $N = n + i$, there are more values of K that lead to a stable system:

- if K is increased, the utilization of the checkout-tills increases, while the utilization of the wrapping-tills decreases
- if K is decreased, the opposite behavior is observed

Changing the utilization and the number of tills for a specific service makes QT vary: some configurations have a better utilization and a better distribution of customers among the tills and they register a shorter QT (e.g. with $p = 0.3$ and $N = 8$, the configuration with $K = 4$ is better than the ones with $K = \{3,5\}$, while other values for K make the system instable).

The difference in using the uniform distribution instead of the exponential one is described in the following analysis:

$\overline{AT}$ = 15 s — *Policy* = united-tills



How QT varies in both distributions using different N

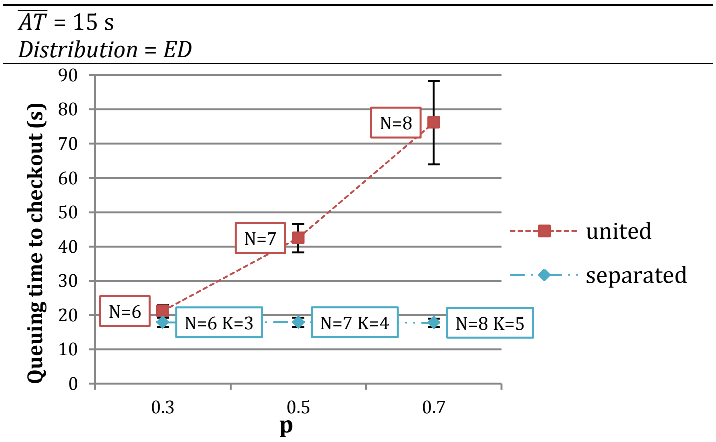
As shown in the graph, the scenarios in *ED* have larger values then the ones in *UD*. The effect is explained by considering different features of the two distributions:

- Uniform random variables have a bounded domain $[min, max]$. Increasing the range for the latter, the mean being equal, has no much impact on $\overline{QT}$. So having $t_x = uniform(15{,}45)$ or $t_x = uniform(0{,}60)$ does not affect performances very much
- Exponential random variables have an infinite domain $[0, +\infty)$. Increasing the left limit (and decreasing its mean accordingly) has an impact on $\overline{QT}$, having that, for example, $t_x = exponential(30)$ generates longer queues than $t_x = 15 + exponential(15)$

These behaviors and the fact that using *UD* $\overline{QT}$ is shorter are due to the shape of the distributions:

- In *UD*, values in the range have the same probability to appear, meaning that periods where t_x is longer are more likely (with respect to *ED*) to be balanced by a longer AT, maintaining the queue stable
- In *ED*, when t_x is longer, there is less probability (with respect to *UD*) that also AT is longer. In this situation the queue tends to increase (as $\overline{QT}$ does) more than in the other distribution. As the longer arrival times are less likely, the queues are more hardly emptied once they are formed (always w.r.t. *UD*). Increasing the left limit of the exponential variables up to min has the result of not having services that will take less than min s. This may lead to think that $\overline{QT}$ may increase because all the $t_x < min$ are excluded (faster services are not allowed) but the effect of decreasing the mean prevails: values generated by $15 + exponential(15)$ have higher probability to be in $[15{,}30]$ than the values generated by $exponential(30)$ to be in $[0{,}30]$. So it is more likely to have faster services and shorter $\overline{QT}$ with the first setting

Is the *separated-tills* policy so bad? By looking at its performances, speaking about the mean queuing time, this policy is worse than the other one. However, by separating the queues for two different services it is possible to give more importance to one of them: if a customer does not need to wrap his/her good but he/she has to wait for others to have their gift ready, he/she might be really annoyed. With *separated-tills* policy it is possible to allocate the right number of tills for a service in order to have less queue for the latter.



How QT varies in both policies for who checks out

The chart indicates that when the wrap service is highly requested, by allocating always 3 tills for checkout in *separated-tills*, those who only need to checkout do not experience longer queue. The same does not happen in the other policy.

Conclusions

United-tills policy is better in terms of tills utilization and distribution of the workload. It has shorter queuing time mean, other factors being equal. However with *separated-tills* policy it is possible to privilege a service: this may cost the usage of more tills but who does not need to wrap his/her good undergoes a much shorter queue (and this is good for the shop reputation).