

**It Is Not How Long It Is, But How You Make It Long –
Waiting Lines in a Multi-step Service Process
(Forthcoming in *System Dynamics Review* 17(4), 2001)**

K. K. Fung

Dept of Economics, U of Memphis, Memphis, TN 38152

Genesis. I have always been interested in waiting lines because waiting in line for me is like hell on earth. I take great pains to avoid waiting lines. My academic interest in queues, however, started when I tried to simulate waiting lines in a closed system such as a ski resort. Little did I know that I was trespassing on hostile territories. It turned out that queuing theory is a very well developed field with voluminous literature. It is hopeless to go through all the literature just to find out what has not been covered yet.

All was not lost. A casual reading of the voluminous literature on queuing theory taught me that a queue must be present even in a steady-state service process. I then inferred that the greater the flow capacity (number of customers served per unit time), the longer must be the steady-state queue if the service process is not to be interrupted. And the minimum steady-state queue should not be regarded as a waiting line at all. Hence the title of this paper.

I also noticed that Naor's seminal paper that started the whole literature on regulating queue length with prices was based on a single-step service process. And this assumption was followed by almost all later contributors. Since I just finished a simulation on Goldratt's scout hike, I knew well that the most binding bottleneck in a multiple-link process does not always coincide with the most severe visible congestion. Instead, the location of the most visible congestion depends on how the links with different flow capacities are positioned. So Naor's observations relating to the effectiveness of regulating queue length with prices may not make much sense in a multiple-step service process where the visible standing queue may be only part of a much longer queue.

Abstract. Accurate signals are essential to ensure the integrity of the feedback mechanism in systems. But in a multi-step service process such as a restaurant business where the flow capacities of the service steps are not identical, visible signals may not indicate the true magnitude to which the system should respond. Specifically, the visible standing queue may not reflect the full length of the waiting line. And if potential customers cannot accurately judge the actual length of the waiting line, varying meal prices to regulate queue length would not be an effective feedback mechanism. Instead, queue length can be more effectively controlled by correctly identifying and widening the most binding bottleneck.

What Your See May Not Be What You Get

In system dynamics, we often implicitly assume that the visible indicators correctly reflect the underlying phenomena. When what you see is what you get, the feedback mechanism in a system will be working perfectly. For example, if the visible queue correctly indicates the length of the whole waiting line for a particular service, service users can decide whether it is worth their while to join the queue. In a seminal paper, Naor (1969) suggested using fee to regulate queue length at toll stations. Specifically, with the imposition of a toll fee, potential queuers can then decide

whether they want to join the queue or not by comparing their marginal cost (as indicated by queue length and toll fee) with their marginal benefit.

Queue length is easy to eyeball in a single step process as in Naor's toll station. But in a multi-step process, the visible queue length is often only part of the whole queue. If you have been to a doctor's waiting room, you would appreciate the fact that the number of patients in that room is no indication of how soon you could get to see the doctor. There may be other patients waiting in other rooms out of your sight.

This note will use the restaurant as an example of a multi-step service process to demonstrate that the visible queue may be only part of a much longer waiting line. When what you see is not necessarily what you get, the feedback mechanism of the system is compromised if only the visible indicators are used.

Service Time in a Multi-Step Service Process

In a tollbooth, the flow capacity (μ , i.e., how many customers can be processed per unit time) and the service time ($1/\mu$, i.e., how long it takes to process a customer) are straightforward since only a one-step service process is involved. In a restaurant, the service process consists of a series of successive service steps, each of which may have different flow capacity and service time. For example, arriving customers must be seated, their orders taken by the waiter, their orders sent to the kitchen, their orders completed in the kitchen and delivered to them, and their meals are eaten and paid for. In other words, there are at least 5 steps between arrival and departure. In between these steps could be separate waiting lines. Most studies (see Stidham, 1985 and Mendelson & Whang, 1990 for comprehensive surveys) are concerned with only the waiting line outside the restaurant. But, in addition to that, there may be at least three waiting lines inside the restaurant. First, there may be a line of seated customers waiting to place their orders. Second, a line of customers waiting to have their orders taken up by the kitchen. And third, a line of customers waiting for their checks after finishing their meals.

In line with every day usage, only those customers who are biding their time that is not part of the core service time are considered to be in a waiting line. Thus, while the cooking time should not be considered part of the waiting time, the long delay between when the orders are placed and when they are taken up by the kitchen should be. In addition, the minimum queue length needed to keep the process going uninterrupted should not be considered as part of the waiting line.

Steady-State in a Multi-Step Service Process

In a single-step service process, a steady-state equilibrium is achieved when arrivals equal departures. If a line is visible, it is sufficiently long only to keep the service process going without interruption. The higher the departure rate, the longer the visible line is. If the constant-length waiting line is longer than just enough to keep the line from disappearing, it is because those in line have already factored in their tolerable waiting time (more later).

In a steady-state multi-step service process, arrivals will still just offset departures with the length of the visible line staying constant. But not only is the length of the line waiting to be seated staying constant, the number of customers distributed over other service points will also stay constant. Their numbers are such that there are just enough customers at each service points to keep the whole process going without interruption.

In a single-step service process, a steady-state waiting line implies that the flow and stock capacity of the service is fully utilized. But in a multi-step service process, unless each of the service components has identical flow capacity, some stock capacity will be underutilized. Only

the service resource that has the lowest flow capacity will be fully utilized.

For example, if the kitchen can finish processing 2 orders per minute while the waiter can take 4 orders per minute and customers can finish eating 4 cooked orders per minute, the steady-state arrival rate and the steady-state departure rate cannot exceed 2 customers per minute. At a flow rate of 2 orders per minute, the kitchen will be fully occupied, but excess stock capacity in waiters and seats will result.

Where Is the Waiting Line? – When the Kitchen Is the Bottleneck

In a single-step service process with one server and no enclosed waiting area such as a tollbooth, the waiting line is visible to outside observers. In a multi-step service process such as a restaurant where the slowest flow capacity is in the kitchen, the number of excess seats may be large enough to partially or totally conceal the line otherwise visible to outside observers as waiting to be seated. In other words, some or all the customers waiting for seats in a restaurant with no excess seats in a steady state would actually be seated in an otherwise identical restaurant with excess seat capacity in a steady state. A passerby looking for a place to eat lunch would have insufficient visual clue as to how long the waiting line is and whether he has enough time to patronize the restaurant. An astute observer who peeks inside the restaurant could tell that the waiting line might already have been too long for him. The telling clue is, of course, the small percentage of the seated customers that are actually eating their meals.

The service process of a restaurant with excess seat capacity in a steady state can be set up as follows:

Parameters (see column 1 of Table 1):

Assumptions

- The cooking process and the eating process are modeled as conveyor belts.
- Each order is assumed to take the same amount of cooking time and eating time.
- Each seat is separately re-configurable into groups to avoid space wastage.
- Each customer has a separate order.
- Each customer has identical benefits and costs (including waiting costs) from the meal.

In this model, the kitchen flow capacity is the binding bottleneck because it (at 2 orders per minute) has the lowest value among all flow capacities. So its flow capacity is analogous to the μ in a single-step service process. If customers keep arriving at 3 per minute, a waiting line will form and lengthen since the departure rate must necessarily be the same as the serving rate from the kitchen (at 2 orders per minute).

But the visibility of the line waiting to be seated depends on how arriving customers are seated. If they are seated no faster than the kitchen can process their orders and must wait outside the restaurant before they are seated, the waiting line will become visible very early (see number of customers awaiting seats in Figure 1A). On the other hand, if customers are seated when empty seats are available, no waiting line of standing customers will be visible until all the seats are filled at the 45th time period (see Figure 1B). But since orders are taken no faster than the kitchen can process them, the waiting line for seats is simply converted into a waiting line to order. This waiting line to order can easily be converted into a line of orders waiting to be taken up by the kitchen if orders are taken at the maximum order-taking flow capacity (not shown). So the *defacto* waiting line (customers awaiting seats plus waiting to order) in Figure 1B is identical to the standing line awaiting seats in Figure 1A.

Where Is the Waiting Line? – When Seats Are the Bottleneck

What happens if the binding flow capacity lies with seats rather than with the kitchen? Suppose every model specification stays the same as above except for the arrival rate and the flow capacity of seats and the kitchen (see column 2 of Table 1):

Here, the kitchen flow capacity matches the arrival rate, but exceeds the binding flow capacity of seats. If customers are seated at the same rate as the binding flow capacity of seats and customers waiting for seats must stay outside the restaurant, a standing line will form and lengthen early. Seats will not be fully occupied until the 36th time period (see Figure 2A). And kitchen capacity will never be fully utilized.

On the other hand, if arriving customers are seated when empty seats are available and orders are processed at the same rate as the flow capacity of the kitchen, the standing line will be shorter and more meals will be served. But since meals are served at a faster rate than the flow capacity of seats, all seats will eventually be occupied by eating customers before any departure begins. Until departure begins, no more customers can be seated and no more orders will go into the totally idle kitchen (see Figure 2B). Such stops-and-starts cycles may be graphically messy, but they can process a lot more customers (68 vs. 48) within the short lunch time window than the smooth and uninterrupted steady-state processes within the restaurant.

How Long Is Too Long?

The two models above demonstrate how difficult it is for the casual observer and potential customers to tell whether there is a waiting line and how long it is. Suppose the potential customer is willing to wait no more than half of the time that it would take to finish the core restaurant process without any waiting, a maximum line can be computed for each of these models. This is the maximum number of customers waiting for seats that a potential customer is willing to put up with if waiting is not concealed (see definition in Table 1 notes). Figure 1 shows that when arriving customers are seated if seats are available, the number of customers waiting for seats would not reach the maximum line until the 52nd time period. But if arriving customers are seated only at the flow capacity of the binding bottleneck (i.e., the kitchen), the maximum line would be reached at the 13th time period. Similarly, Figure 2 shows that when customers are seated if seats are available, the number of customers waiting for seats would not reach the maximum line until the 22nd time period. But if customers are seated only at the flow capacity of seats (the binding bottleneck here), the maximum line would have been reached at the 10th time period.

Because the waiting line for seats can be easily concealed, potential customers cannot easily judge whether the waiting line is too long even though they have a definite idea of how long they are prepared to wait. Many of them end up waiting much too long.

Shortening the Line – Non-pricing System Dynamics Approach

The above simulations have demonstrated how easy it is to conceal the true magnitude of a visible signal in the form of waiting lines. The general message of this exercise to system designers is quite simple. Namely, if they rely on only visible signals to design the feedback mechanism, system operation could be severely compromised (Homer, 1999). Instead, whenever a multi-step process is involved, they should look beyond visible congestion to correctly identify the most binding bottleneck (Fung, 1999). It is this most binding bottleneck that the feedback mechanism of the system should be based on if the system is to operate efficiently.

In the current example, the inability of potential customers to accurately judge the full

length of the waiting line makes meal prices a poor instrument to fine tune the length of waiting lines. If, however, the kitchen is determined to be the binding bottleneck (see Figure 1), increasing the kitchen staff or increasing the productivity of the kitchen staff will increase the flow capacity of the process and effectively shorten the waiting line.

When the flow capacity of seats is the bottleneck, the obvious solution is to increase the number of seats. Renting more space adjacent to the restaurant can do this or seats can be redesigned to accommodate more customers within the same space. Flow capacity of seats can be increased not only by increasing the number of seats, but also by reducing eating time. For example, if eating time can be reduced by 1/3 in Figure 2B, the number of customers completing their lunches would increase by 57% (not shown in Figure 2). Here, more ingenuity is required. The most common approach is for waiters to subtly suggest that the meal be concluded.

The advantages of the non-pricing approach are obvious. It leaves the arrival rate well alone because it is inherently unpredictable (Becker, 1991). Instead, it works on the turnover rate to accommodate as many of the arriving customers as possible by increasing the flow capacity of the restaurant process. And the adjustment is in the hands of the restaurant management who should have the best information on the flow capacity of each service component, where the binding bottleneck is, and how long the waiting line actually is.

References

- Becker, G. S. 1991. A Note on Restaurant Pricing and Other Examples of Social Influences on Price. *Journal of Political Economy* **99**(5): 1109-1116.
- Fung, K. K. 1999. Follow the Laggard? – Not all Bottlenecks are Created Equal. *System Dynamics Review* **15**(4): 403-410.
- Homer, J. 1999. Macro- and Micro-modeling of Field Service Dynamics. *System Dynamics Review* **15**(2):139-162.
- Mendelson, H. & S. Whang. 1990. Optimal Incentive-Compatible Priority Pricing for the M/M/1 Queue. *Operations Research* **38**(5): 870-883.
- Naor, P. 1969. The Regulation of Queue Size by Levying Tolls. *Econometrica* **37**(1): 15-24.
- Stidham, S. Jr. 1985. Optimal Control of Admission to a Queuing System. *IEEE Transactions on Automated Control* **41**(1): 163-173.

Notes

- This work was supported in part by a grant from the Fogelman College of Business and Economics at the University of Memphis. This research support does not necessarily imply endorsement of the research results by either the Fogelman College or the University of Memphis.
- *Stella* model diagrams and equations may be requested from kkfung@memphis.edu

Table 1: Parameters of Two Multi-step Restaurant Processes

	Kitchen too slow	Seats too few
<i>Flow Capacity</i>		
Seat turnover flow capacity [$\mu_1 = \text{integer}(N_1/T_1)$]	6/min	2/min
Kitchen flow capacity ($\mu_2 = N_2/T_2$)	2/min	4/min
Order taking flow capacity ($\mu_3 = N_3/T_3$)	4/min	4/min
Order sending flow capacity ($\mu_4 = N_4/T_4$)	4/min	4/min
Arrival rate (λ)	3/min	4/min
<i>Stock capacity</i>		
Number of seats (N_1)	68	68
Kitchen capacity (N_2)	6 orders	8 orders
Order taking waiters (N_3)	4	4
Order sending waiters (N_4)	4	4
<i>Other parameters</i>		
Eating time per order (T_1)	10 min	30 min
Cooking time per order (T_2)	3 min	2 min
Order taking time (T_3)	1 min	1 min
Order sending time (T_4)	1 min	1 min
<i>Max tolerable line length</i>		
	15 queuers	34 queuers

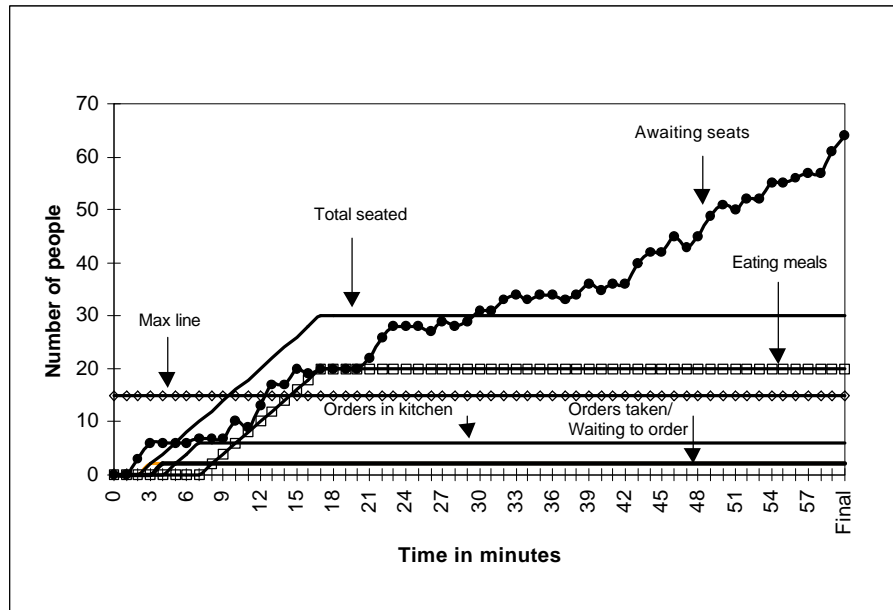
Notes:

Maximum tolerable line length = Maximum waiting time * bottleneck flow capacity.

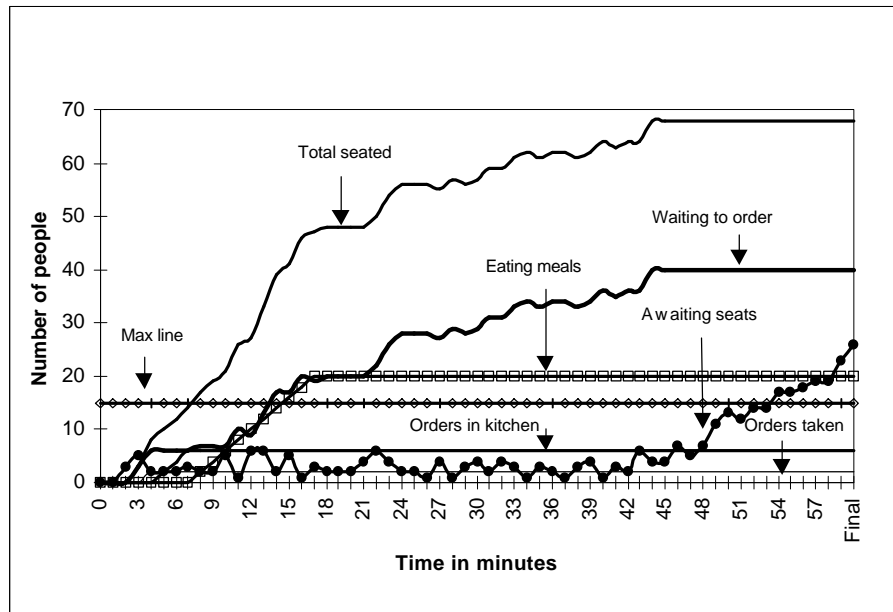
If maximum tolerable waiting time is $\frac{1}{2}$ of total service time, then maximum tolerable line length
 = $(T_1 + T_2 + T_3 + T_4)/2$ * bottleneck flow capacity.

Figure 1: Restaurant Waiting Line with Kitchen as Bottleneck

A. Customers are seated at kitchen flow capacity



B. Customers are seated as seats become available

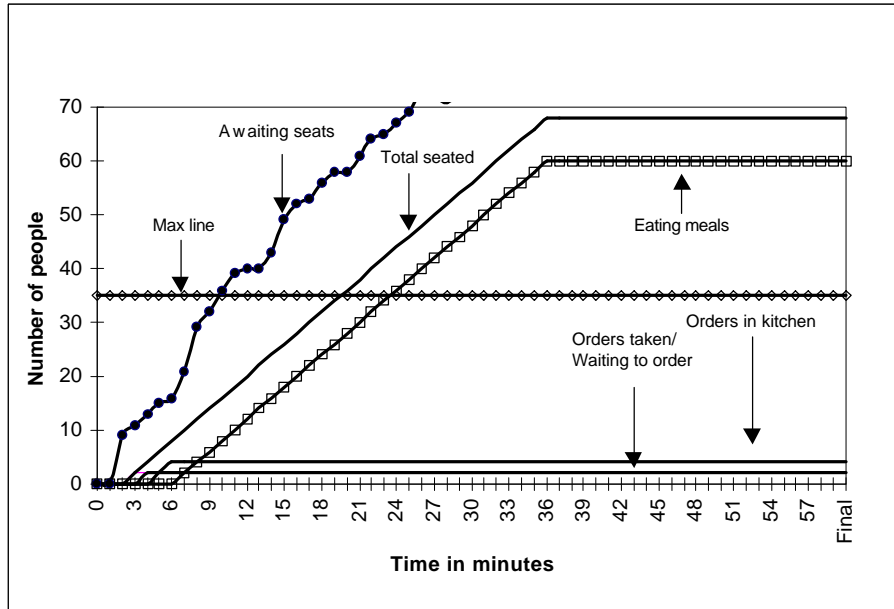


Notes:

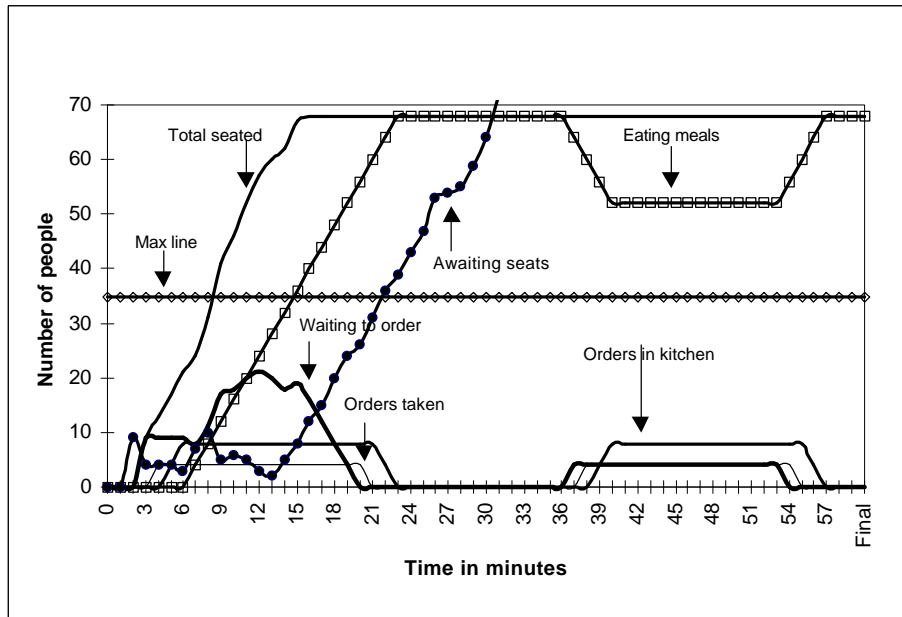
- Arrival rate exceeds kitchen flow capacity but below flow capacity of seats.
- No waiting line is visible outside the restaurant until the 45th time period in Figure 1B when all seats are fully occupied. But waiting line is visible almost from the start in Figure 1A.

Figure 2: Restaurant Waiting Line with Seats as Bottleneck

A. Customers are seated only at flow capacity of seats



B. Customers are seated as seats become available



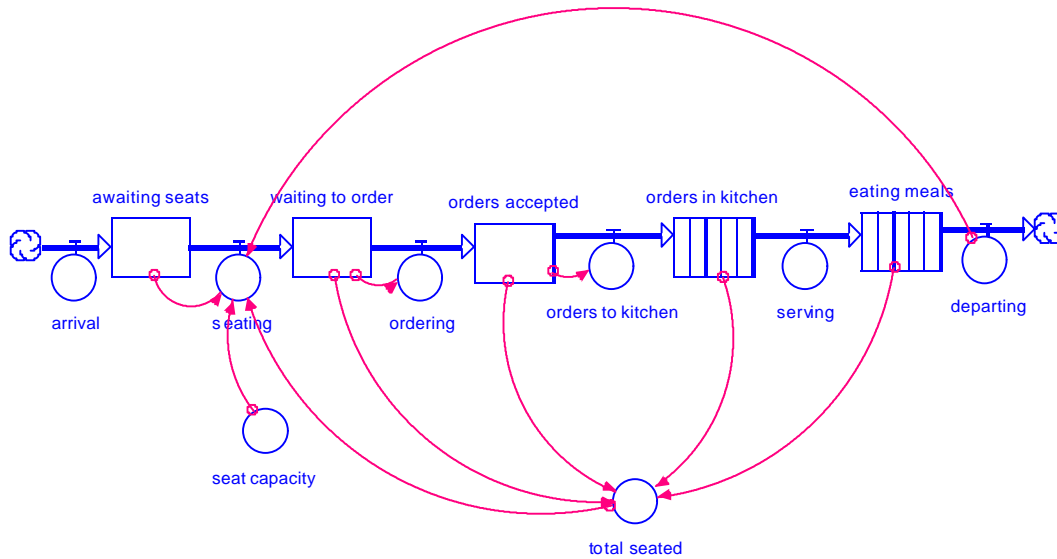
Notes:







- The binding bottleneck of seats is located after the kitchen.
- More customers (68 vs. 48) can be processed in the lunch time window when orders are taken and sent to kitchen at the kitchen flow capacity in Figure 2B than in Figure 2A. But the waiting line to order conceals the waiting line for seats.

Appendix1

Restaurant Waiting Line – Customers Are Seated at Bottleneck Flow Capacity

Stella Model Diagram and Equations for Figure 1A



- $\text{awaiting_seats}(t) = \text{awaiting_seats}(t - dt) + (\text{arrival} - \text{seating}) * dt$
 INIT awaiting_seats = 0
 INFLOWS:
 arrival = POISSON(3, 123)
 OUTFLOWS:
 seating = IF (total_seated = seat_capacity) THEN MIN(departing, awaiting_seats)
 ELSE IF (total_seated < seat_capacity)
 THEN MIN (awaiting_seats, seat_capacity - total_seated + departing)
 ELSE 0
- $\text{waiting_to_order}(t) = \text{waiting_to_order}(t - dt) + (\text{seating} - \text{ordering}) * dt$
 INIT waiting_to_order = 0
 INFLOWS:
 seating = IF(total_seated=seat_capacity) THEN MIN(departing, awaiting_seats)
 ELSE IF(total_seated<seat_capacity)
 THEN MIN(awaiting_seats, seat_capacity - total_seated + departing) ELSE 0
 OUTFLOWS:
 ordering = MIN(2, waiting_to_order)
- $\text{orders_accepted}(t) = \text{orders_accepted}(t - dt) + (\text{ordering} - \text{orders_to_kitchen}) * dt$
 INIT orders_accepted = 0
 INFLOWS:
 ordering = MIN(2, waiting_to_order)
 OUTFLOWS:
 orders_to_kitchen = MIN(2, orders_accepted)

$$\text{orders_in_kitchen}(t) = \text{orders_in_kitchen}(t - dt) + (\text{orders_to_kitchen} - \text{serving}) * dt$$
 INIT orders_in_kitchen = 0
 TRANSIT TIME = 3
 INFLOW LIMIT = 2
 CAPACITY = 6
 INFLOWS:

$$\text{orders_to_kitchen} = \text{MIN}(2, \text{orders_accepted})$$
 OUTFLOWS:

$$\text{serving} = \text{CONVEYOR OUTFLOW}$$

$$\text{eating_meals}(t) = \text{eating_meals}(t - dt) + (\text{serving} - \text{departing}) * dt$$
 INIT eating_meals = 0
 TRANSIT TIME = 10
 INFLOW LIMIT = INF
 CAPACITY = INF
 INFLOWS:

$$\text{serving} = \text{CONVEYOR OUTFLOW}$$
 OUTFLOWS:

$$\text{departing} = \text{CONVEYOR OUTFLOW}$$

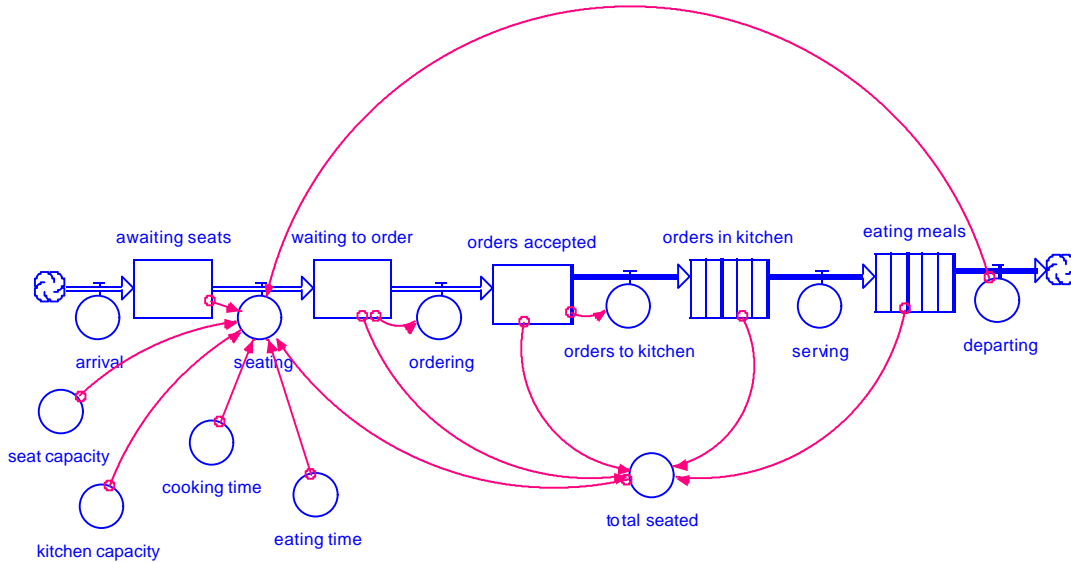
- seat_capacity = 68
- total_seated = orders_accepted + eating_meals + waiting_to_order + orders_in_kitchen

N.B.: Same model and equations for Figure 2A except for value changes of parameters as listed in Table 1 column 2.

Appendix 2

Restaurant Waiting Line – Customers Are When Seats Are Available

Stella Model and Equations for Figure 1B



$$\square \text{ awaiting_seats}(t) = \text{awaiting_seats}(t - dt) + (\text{arrival} - \text{seating}) * dt$$

INIT awaiting_seats = 0

INFLOWS:

$$\overrightarrow{\text{arrival}} = \text{POISSON}(3, 123)$$

OUTFLOWS:

$$\overrightarrow{\text{seating}} = \text{IF}(\text{total_seated} = \text{seat_capacity}) \text{ THEN } \text{MIN}(\text{departing}, \text{awaiting_seats})$$

$$\text{ELSE IF}(\text{total_seated} < \text{seat_capacity})$$

$$\text{ THEN } \text{MIN}(\text{awaiting_seats}, \text{INT}(\text{seat_capacity}/\text{eating_time}),$$

$$\text{INT}(\text{kitchen_capacity}/\text{cooking_time}))$$

$$\text{ ELSE } 0$$

$$\square \text{ waiting_to_order}(t) = \text{waiting_to_order}(t - dt) + (\text{seating} - \text{ordering}) * dt$$

INIT waiting_to_order = 0

INFLOWS:

$$\overrightarrow{\text{seating}} = \text{IF}(\text{total_seated} = \text{seat_capacity}) \text{ THEN } \text{MIN}(\text{departing}, \text{awaiting_seats})$$

$$\text{ ELSE IF}(\text{total_seated} < \text{seat_capacity})$$

$$\text{ THEN } \text{MIN}(\text{awaiting_seats}, \text{INT}(\text{seat_capacity}/\text{eating_time}),$$

$$\text{INT}(\text{kitchen_capacity}/\text{cooking_time}))$$

$$\text{ ELSE } 0$$

OUTFLOWS:

$$\overrightarrow{\text{ordering}} = \text{waiting_to_order}$$

□ orders_accepted(t) = orders_accepted(t - dt) + (ordering - orders_to_kitchen) * dt
 INIT orders_accepted = 0
 INFLOWS:
 → ordering = waiting_to_order
 OUTFLOWS:
 → orders_to_kitchen = orders_accepted

▣ orders_in_kitchen(t) = orders_in_kitchen(t - dt) + (orders_to_kitchen - serving) * dt
 INIT orders_in_kitchen = 0
 TRANSIT TIME = 3
 INFLOW LIMIT = 2
 CAPACITY = 6
 INFLOWS:
 → orders_to_kitchen = orders_accepted
 OUTFLOWS:
 → serving = CONVEYOR OUTFLOW

▣ eating_meals(t) = eating_meals(t - dt) + (serving - departing) * dt
 INIT eating_meals = 0
 TRANSIT TIME = 10
 INFLOW LIMIT = INF
 CAPACITY = INF
 INFLOWS:
 → serving = CONVEYOR OUTFLOW
 OUTFLOWS:
 → departing = CONVEYOR OUTFLOW

- cooking_time = 3
- eating_time = 10
- kitchen_capacity = 6
- seat_capacity = 68
- total_seated = orders_accepted+eating_meals+waiting_to_order+orders_in_kitchen

N.B.: Same model and equations for Figure 2B except for value changes of parameters as listed in Table 1, column 2.