Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

ESTIMATING AND INTERPRETING THE WAITING TIME FOR CUSTOMERS ARRIVING TO A NON-STATIONARY QUEUEING SYSTEM

Jeffrey S. Smith

Department of Industrial and Systems Engineering
Auburn University
3301 Shelby Center
Auburn, AL 36849, USA

Barry L. Nelson

Department of Industrial Engineering and Management Sciences
Northwestern University
2145 Sheridan Road
Evanston, IL 60208, USA

ABSTRACT

When a customer arrives to a service system, how long should they expect to wait, and how long might their wait actually be? Computer simulation is an ideal tool for answering such questions for very general and complex queueing systems, but they are not always answered by the automatic statistical summary generated by commercial simulation languages. Using an illustration based on passenger check-in at an airport, we demonstrate how standard summary measures go wrong and provide methods that correctly answer these questions.

1 INTRODUCTION

We focus on assessing and interpreting the *virtual waiting time* of customers arriving to a queueing system that is not necessarily in "steady state." Stated simply, the virtual waiting time is the waiting time that an arriving entity would expect to see given that the entity arrives at a particular (simulated) time. We show that this measure is particularly interesting in a variety of systems and that many "standard" simulation experiment metrics appear to measure this, but do not, often leading an incorrect interpretation by the user.

We start with a typical service system where the virtual waiting time will be of specific interest – passengers arriving to an airport check-in counter where an important question is "how long will it take me to check-in for my flight?" Even more specifically, a passenger with a 9:00 AM flight may want to know "Am I likely to be checked in by 8:30 if I arrive at 7:30?" Clearly this time estimate is important in determining when a passenger should leave their house to make their flight – underestimating this time can have dire consequences. Answering this question requires an assessment of the distribution of wait time for a customer arriving at 7:30 AM; in queueing theory this is called the virtual waiting time (Gross *et al*., 2008). Generically, let $W(t)$ be the virtual waiting time for a customer arriving at time t. Of course, while the customer is focused on arriving at a specific time, airport management may be interested in $W(t)$ throughout the day, not just at t = 7:30 AM. Possible summary measures of the distribution of $W(t)$ include its mean, standard deviation and extreme percentiles.

Our simplified airport check-in system will have the following characteristics:

- The model is terminating – Arrivals start at 6:00 a.m., end at 10:00 p.m., and the system stays open until all passengers are processed;
- The passenger arrival process is non-stationary;
- The check-in counter has time-dependent agent capacity (using a resource schedule) and a single passenger queue;

- Premium and Regular passengers – the passenger queue is prioritized based on passenger type.

While there are clear interests in system performance from the airport management perspective (overall system performance metrics), we focus exclusively on the arriving passenger's perspective in this paper. In particular, we wish to use a simulation model to estimate the virtual waiting time for a passenger arriving at a given time. We use a Simio simulation (described in the next section) for our analysis and will show how the "standard" measures of time-in-system (*TIS*) can be misleading and how to compute more appropriate estimates of the virtual waiting times. Other simulation packages that we are familiar with generate very similar performance metrics/statistics – the issues we highlight are not Simio-specific.

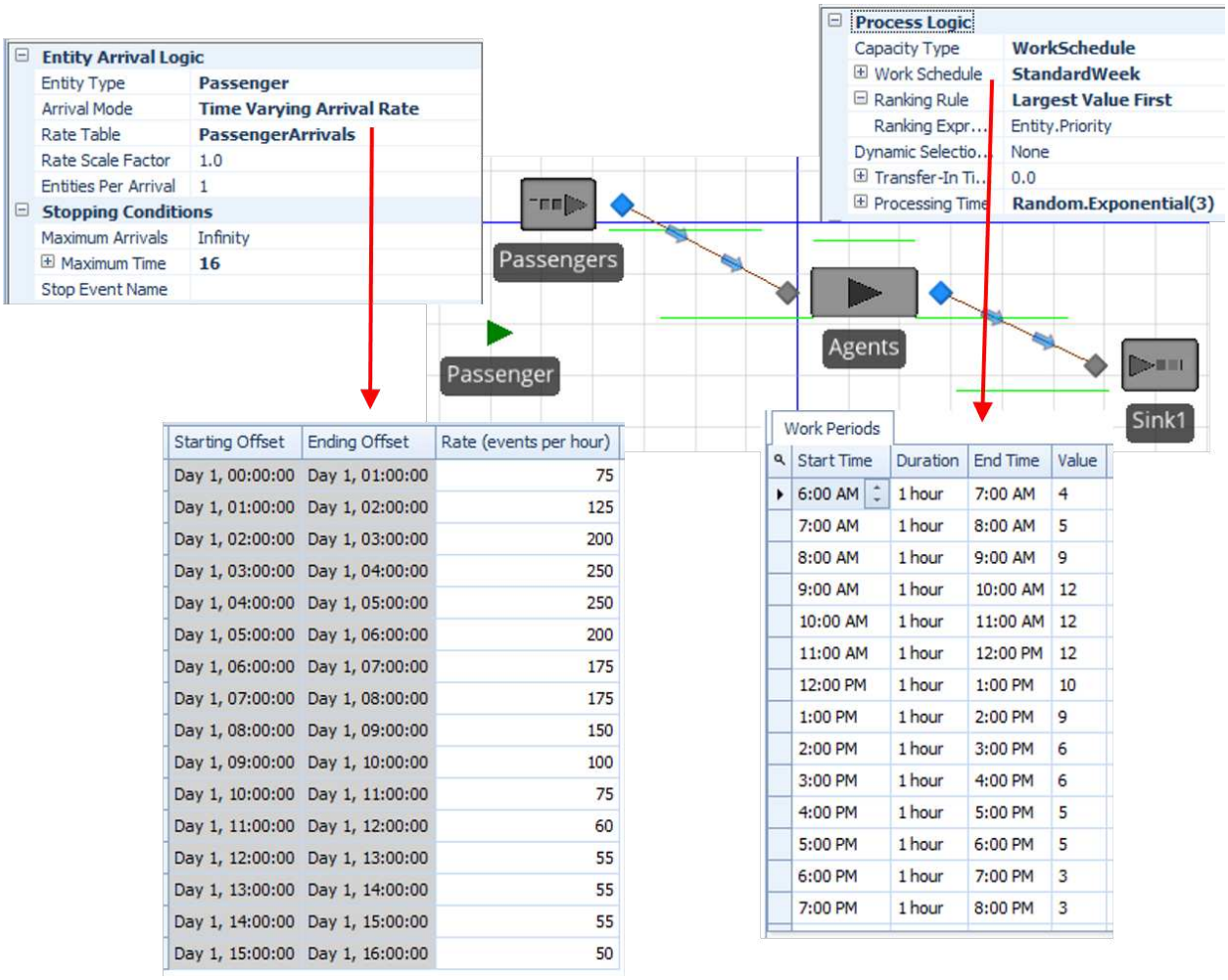
2 INITIAL MODEL

Figure 1 illustrates our basic model. The model is a single-server queueing system with a non-stationary arrival process and a time-dependent Agent capacity (using a resource schedule). This mimics the basic behavior of the "ticket counter" at many airports. The object properties that define the arrival process and the resource schedule are also illustrated in the figure for clarity. The prioritization by passenger type is handled using the Ranking Rule for the Agents object instance (Largest Value First of Entity.Priority). The passenger type for an arriving entity is set probabilistically using an add-on process triggered on entity creation in the Passengers object instance (the add-on process is not shown in Figure 1).

The model collects standard performance metrics including the number of passengers in the queue and the times that passengers (overall and by passenger type) spend in the system. Figure 2 shows a dynamic status plot for one replication of the model. The plot shows the instantaneous number of passengers in the queue (NIQ) and the cumulative average time passengers spend in the system by passenger type and overall (TISPremium, TISRegular, and TISOverall, respectively). Finally, Figure 3 shows the standard SMORE (Simio MORE) plot (Nelson, 2008) for the average time that regular customers spend in the system based on 500 replications of the model. We chose these specific metrics/performance measures because they would be "standard" results computed "at no extra cost to the modeler" by the Simio model (note that similar standard statistics are computed automatically by many other commercial simulation packages). As such, these would be the likely metrics that users would initially examine to assess the system performance characteristics.

There is nothing inherently "wrong" with these performance measures/plots if interpreted correctly; we simply suggest that they provide almost no useful information (and perhaps even misleading information) about our performance metric of interest – the virtual waiting time. To make an informed choice about when to leave for the airport, a passenger needs good information about the expected time in system, the standard deviation of the time in system, and perhaps extreme percentiles for the corresponding distribution of time in system *for the specific time that the passenger arrives to the airport*. A quick glance at the NIQ line in Figure 2 shows that the number of passengers waiting in line varies dramatically throughout the day. In our simplified system, the time that passengers spend in the system will similarly vary throughout the day. However, the TISRegular, TISPremium, and TISOveall lines in the plot do not exhibit this variation by the nature of how they are computed (each is a "running average" through time).

The SMORE (Simio Measure of Risk and Error) plot (Figure 3) is an excellent tool for controlling the sampling error (using the confidence intervals) to determine the appropriate number of replications to confidently assess the distribution characteristics (the mean, variance, percentiles, etc.). Accordingly, one may be tempted to use these distribution characteristics to answer the "when should I leave for the airport" question. However, the SMORE plot provides good information about the *mean* passenger time in the system *over the entire replication* which corresponds to one day in our example. In more detail, the SMORE Plot is based on the following: On replication $j = 1,2, \ldots, n$ of the simulation, let $W_{1j}, W_{2j}, \ldots W_{N_j j}$ be the customer waiting times in the order in which customers complete their waiting.



Starting Offset	Ending Offset	Rate (events per hour)
Day 1, 00:00:00	Day 1, 01:00:00	75
Day 1, 01:00:00	Day 1, 02:00:00	125
Day 1, 02:00:00	Day 1, 03:00:00	200
Day 1, 03:00:00	Day 1, 04:00:00	250
Day 1, 04:00:00	Day 1, 05:00:00	250
Day 1, 05:00:00	Day 1, 06:00:00	200
Day 1, 06:00:00	Day 1, 07:00:00	175
Day 1, 07:00:00	Day 1, 08:00:00	175
Day 1, 08:00:00	Day 1, 09:00:00	150
Day 1, 09:00:00	Day 1, 10:00:00	100
Day 1, 10:00:00	Day 1, 11:00:00	75
Day 1, 11:00:00	Day 1, 12:00:00	60
Day 1, 12:00:00	Day 1, 13:00:00	55
Day 1, 13:00:00	Day 1, 14:00:00	55
Day 1, 14:00:00	Day 1, 15:00:00	55
Day 1, 15:00:00	Day 1, 16:00:00	50

Figure 1. Initial Simio model.

For each replication, compute the replication mean $\overline{W}_j$ – these replication means provide the data for the SMORE plot. The sample mean of the daily average waiting times is given by

$$\overline{W} = \frac{1}{n}\sum_{j=1}^{n}\overline{W}_j = \frac{1}{n}\sum_{j=1}^{n}\frac{1}{N_j}\sum_{i=1}^{N_j}W_{ij}$$

This is the brownish dot around the value of 21 in Figure 3 and this value is commonly computed in other simulation languages/packages as well. This is typically accompanied by a confidence interval (CI) of the form

$$\overline{W} \pm t_{1-\frac{\alpha}{2},n-1}\frac{S}{\sqrt{n}}$$

where S is the standard deviation of the daily averages $\overline{W}_1, \overline{W}_2, \ldots \overline{W}_n$. This is the brownish box surrounding the sample mean in Figure 3.

While one might be tempted to use these measures to answer the question of when to depart for the airport, none of them are useful for the following reasons:

- $\overline{W}$ does not estimate the mean, or expected value, of waiting time for an arrival at 7:30 AM (or any other fixed time) *unless we believe that the waiting time process is stationary*. If the arrival rates, staffing levels, or check-in protocols change throughout the day, then the process is not stationary. The data $\overline{W}_1, \overline{W}_2, \ldots \overline{W}_n$ are averages *through time* (from the start to the end of the check-in day) and therefore they mask the time-dependent effects (see the NIQ line in Figure 2).
- The CI is a *measure of error* for how well $\overline{W}$ estimates its true mean E($\overline{W}$). You can think of the ``true mean'' as what $\overline{W}$ would become if you let the number of replications n grow infinitely large; the CI measures the error from stopping with finite n. Thus, the CI does not tell us anything about the variability of an individual customer's check-in experience – a critical factor for determining an appropriate departure time.
- Although the standard deviation S is a measure of variability, it is the variability of the *daily averages* $\overline{W}_1, \overline{W}_2, \ldots \overline{W}_n$ not the individual customer's waiting times. So it also is not the right answer.

Figure 2. Status plot for Number-in-Queue and Time-In-System (in minutes).

3 AN ALTERNATIVE APPROACH

It is conceptually easy to simulate the random variable $W(t)$ for some specific time like t = 7:30 AM: On each of n replications, insert a marked customer arrival exactly at 7:30 AM, track the customer until they clear check-in, and record their waiting time. Let W_1 (7:30),…,W_n (7:30) be the observed waiting times of the marked customer across n replications. These data will be independent and identically distributed (i.i.d.) so standard statistical analysis applies for estimating the mean, standard deviation and percentiles of virtual waiting time. As a practical matter, however, this method has problems: There is the minor programming issue of inserting and tracking a customer arrival exactly at 7:30 AM. More importantly, this method should only be applied for a single fixed time, like 7:30 AM. If we want to assess the virtual waiting time throughout the day, and do so by inserting marked arrivals every, say, 1 minute, then we substantially increase the actual load on the check-in system and the model is no longer a valid representation of the airport. Our focus in this section will be on how to get useful approximations

to the virtual waiting time distribution, or at least the mean and standard deviation of it, without using the "insert an entity" approach.

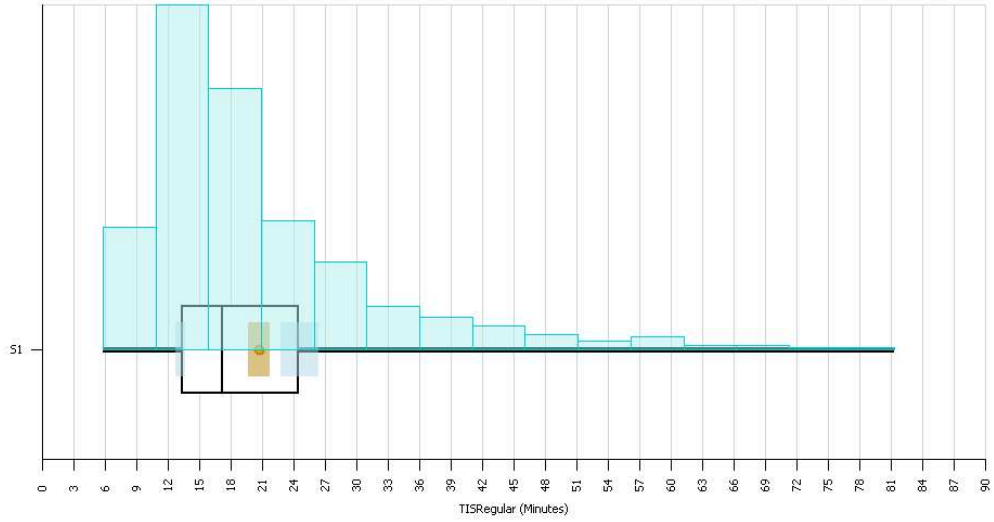


Figure 3. SMORE plot for the average time regular passengers spend in the system.

Two key insights are important: First, the waiting times that are naturally generated by the simulation, $W_{1j}, W_{2j}, \ldots W_{N_j j}$, are waiting times in the order in which waiting is *completed*; therefore they are not directly useful for the virtual waiting time problem. Instead, we need to consider waiting times ordered by the time of customer arrival. Second, to avoid inserting virtual customers or directly calculating virtual waiting time, we can partition time into contiguous intervals or buckets and let arrivals during each time bucket represent what would happen to virtual arrivals during that period. We need these buckets to be short enough that arrivals during the interval see essentially the same system load, but long enough so that it is almost certain that there will be arrivals during each interval (observing no arrival during a time interval does not mean the virtual waiting time is 0 because the system could be highly congested from previous intervals). When the simulation is driven by a nonstationary Poisson arrival process with piecewise constant rate, then a bucket length that corresponds to the length of a constant-arrival-rate interval is a possible starting point and will be our approach in this paper.

From here on we focus on a specific time bucket (say the one that contains 7:30 AM), but this is only for expository convenience – the ideas apply to all of the buckets and our implementation below divides the (simulated) day into individual buckets and processes all of them. To make the notation distinct, let $Y_{1j}, Y_{2j}, \ldots Y_{M_j j}$ be the waiting times on replication j of the customers who *arrived* during this time bucket. Then

$$\bar{Y} = \frac{1}{n}\sum_{j=1}^{n} \bar{Y}_j = \frac{1}{n}\sum_{j=1}^{n} \frac{1}{M_j}\sum_{i=1}^{M_j} Y_{ij}$$

is, by its definition, an unbiased estimator of the expected value of the sample average waiting time for arrivals during the time bucket, and

$$\bar{Y} \pm t_{1-\frac{\alpha}{2},n-1}\frac{S}{\sqrt{n}}$$

is a valid CI, where S is now the standard deviation of the daily bucket averages $\bar{Y}_j$. In a sense, this makes the average value $\bar{W}$, which was previously wrong because it averaged across the entire day, relevant by restricting it to arrivals in a small time bucket – assuming the buckets are set correctly.

We implemented this method in Simio so that our model would generate SMORE plots for each of the time buckets. To do this, we defined a Tally statistic (an observational statistic) for each of the 16 buckets (we choose to use hourly time buckets to match our arrival process) and created a table with 16 rows so that we could access the individual Tally statistics using an index (the table row). Finally, we need to tell the model to store each observation (the TIS for each entity) in the correct Tally statistic (time bucket). The trick here is that we need to use a passenger's *arrival* time rather than *departure* time to determine the appropriate time bucket. The Tally and table definitions and the add-on processes that implement this method are shown in Figure 4 (the add-on process is executed when an entity enters the sink). The Assign step determines the integer hour (0-based) that the entity arrived (HrIndex) – for our model, this will be an integer between 1 and 16. The Tally step records the time that the entity has spent in the system to the correct tally statistic using the HourlyTIS table and the bucket index computed in the previous step.

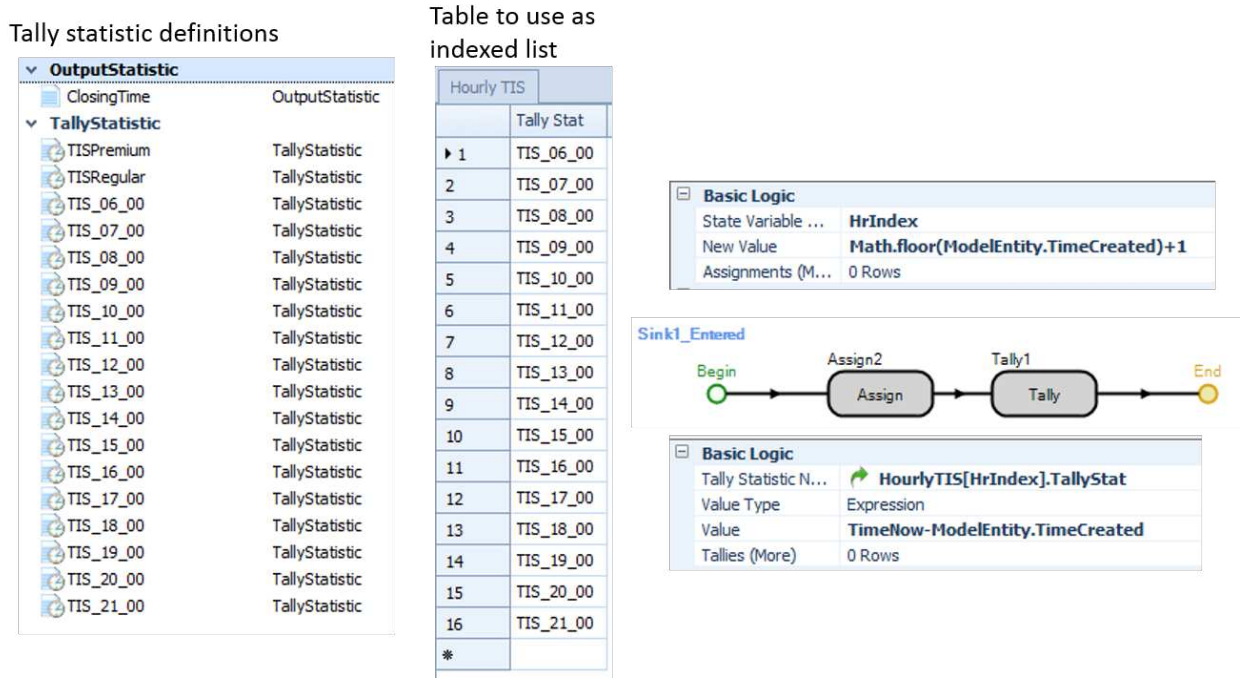


Figure 4. Tabulation of time-in-system by hourly buckets.

Figure 5 shows four of the 16 resulting SMORE plots (we arbitrarily choose the hours from 6:00 a.m. until 10:00 a.m.). The "T_07" bucket includes our 7:30 airport arrival and indicates that the mean check-in time during this hour is approximately 20 minutes. This method will give us good information about the average TIS by hour (we can use replications to make them arbitrarily "good" using the CI as our guide). Of course, the risk-averse traveler will not be comfortable with planning based only on the average delay, even if well estimated, and it is in assessing the *variability* that things become tricky as the standard SMORE plot risk measures are relative to the bucket means rather than to the individual arriving passenger times (as discussed above). While the data within a time bucket are clearly dependent, if we have chosen the time buckets well then we can treat them as approximately stationary, which implies that we can treat the waiting times as identically distributed; this will be key.

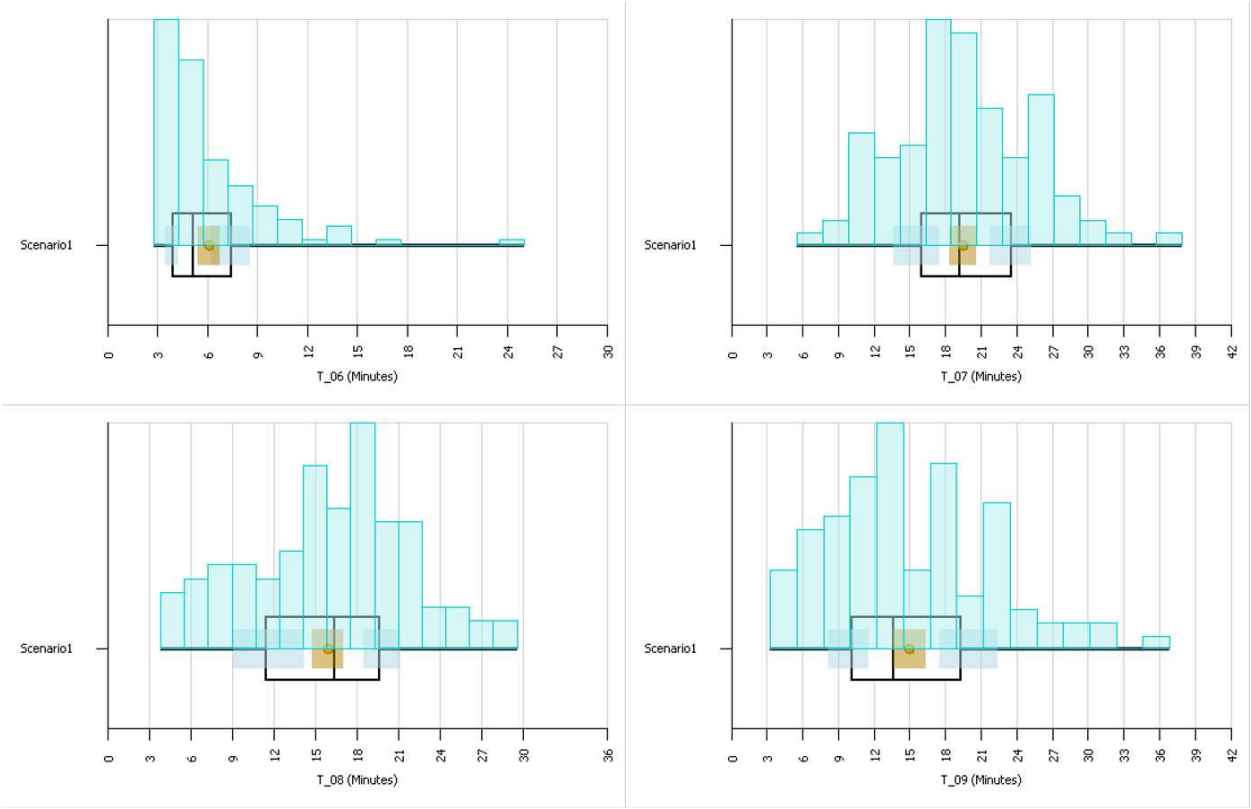


Figure 5. SMORE plots for hourly buckets for 6:00 a.m. - 10:00 a.m.

We develop the variance estimator intuitively by considering the following thought experiment: You have run n replications of the simulation and saved all of the waiting times within the time bucket of interest on each replication. Now you are asked to use these data to *simulate* the delay of an arbitrary customer arriving during that time bucket. Here is an algorithm that makes intuitive (and statistical) sense:

1. Randomly select one of the n replications, say replication J, with equal probability $1/n$.
2. From the M_J waiting times recorded from the time bucket on replication J, randomly select one of them with equal probability $1/M_J$.
3. Return the selected value as $\hat{Y}$.

What is the variance of $\hat{Y}$, $\mathrm{Var}(\hat{Y})$, given the data, for this algorithm?

$$\begin{aligned} \hat{\sigma}^2 = \mathrm{Var}(\hat{Y}|\mathrm{data}) &= \mathrm{E}[\mathrm{Var}(\hat{Y}|J)] + \mathrm{Var}[\mathrm{E}(\hat{Y}|J)] \\ &= \mathrm{E}(S_J^2) + \mathrm{Var}(\bar{Y}_J) \\ &= \frac{1}{n}\sum_{j=1}^{n} S_j^2 + \frac{1}{n}\sum_{j=1}^{n}(\bar{Y}_j - \bar{Y})^2 \end{aligned}$$

where S_j^2 is the sample variance of the waiting times in the time bucket on replication j. Technically, this is the *bootstrap* estimator of the variance of a random waiting time from the time bucket (Shao and Tu, 1995). It consists of two terms: the average variability around the mean of the observations in the bucket, and the variance of the bucket mean itself; its performance might be improved slightly by dividing by $n-1$ instead of n in the second term. This estimator acknowledges that in a smallish interval the mean

waiting time may vary substantially from day to day, and there is likely a strong dependence between the sample mean and sample variance.

Unfortunately, adding this computational logic to Simio is not as straightforward as was the hourly buckets (where we were able to use Simio's built-in SMORE plot generation logic), so we chose to "export" the required data and use an external program to conduct the further analysis. Figure 6 shows the updated add-on process that is executed when entities enter the sink (as previously described, the Assign step determines the time bucket and the Tally step records the TIS to the appropriate Tally – see Figure 4). We added a Write step that writes the individual entity arrival times (`ModelEntity.TimeCreated`) and the TISs (`TimeNow - ModelEntity.TimeCreated`) to an external file.

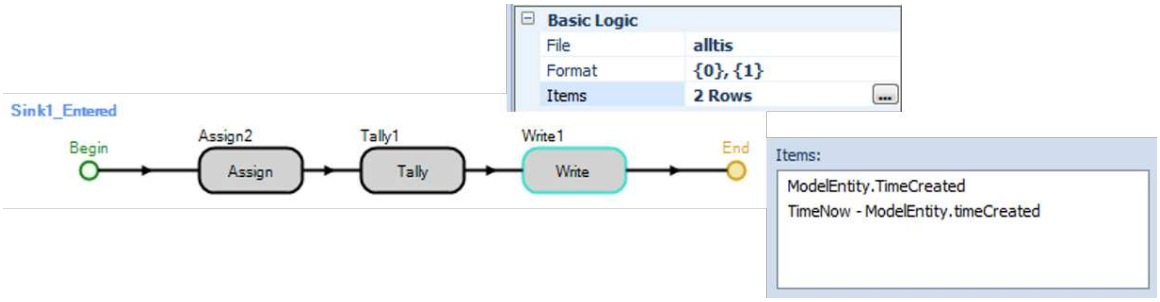


Figure 6. Write step that writes the individual entity arrival times and TISs to an external file.

Next, we ran 500 replications of our model, creating 500 external files – one file for each replication. The files contained the arrival time/TIS pairs for each entity in the corresponding replication. Finally, we used a Python program to aggregate the data from all replications by time bucket. At this point, we had 16 buckets (*lists* in our Python program) where each bucket included the individual TIS values for entities that arrived during the corresponding hour across all 500 replications – the $Y_{1j}, Y_{2j}, \ldots Y_{M_j j}$ values described above for all buckets. Once we had these values, computation of the variance terms as described above was straightforward.

Before discussing the analysis, we make one additional change to the model. Since our Python program had all of the individual Y_{ij} values, we plotted histograms of the individual hourly buckets aggregated across all 500 replications – so each histogram includes *all* observations within the given hour over all replications. Figure 7 shows the histograms for the same 4-hour period as the SMORE plots in Figure 5. We were initially surprised by the bi-modal shape for the 8:00 and 9:00 buckets (and the start of this shape in the 7:00 hour). Some thought and model investigation led us to the conclusion that this is due to the mixing of the Premium and Regular passengers in our Tally statistics. As the system becomes busier, the benefit of being a Premium passenger – prioritization in the check-in line – becomes more pronounced. Since any individual passenger will either be Premium or Regular and will obviously know this when planning their departure time, we need to separate the observations so that we can evaluate the respective virtual waiting time distributions separately. While looking at the histograms in Figure 7 this phenomenon seems obvious (and one could certainly argue that we should have known this before seeing the histograms through careful examination of the plots in Figure 2 and the corresponding experimental results) it is important to note that the bi-modality is definitely not discernable from the standard SMORE plots for the overall TIS or the time-bucket TISs. As such, it could easily be overlooked. In this case, observing the "raw data" in histogram form saved us. Therefore, we modified the Simio model so that it creates separate buckets by passenger type and hour, resulting in 32 individual buckets. In the following analysis, we focus exclusively on the Regular passenger buckets (the Premium passenger analysis would be identical from a methodological standpoint).

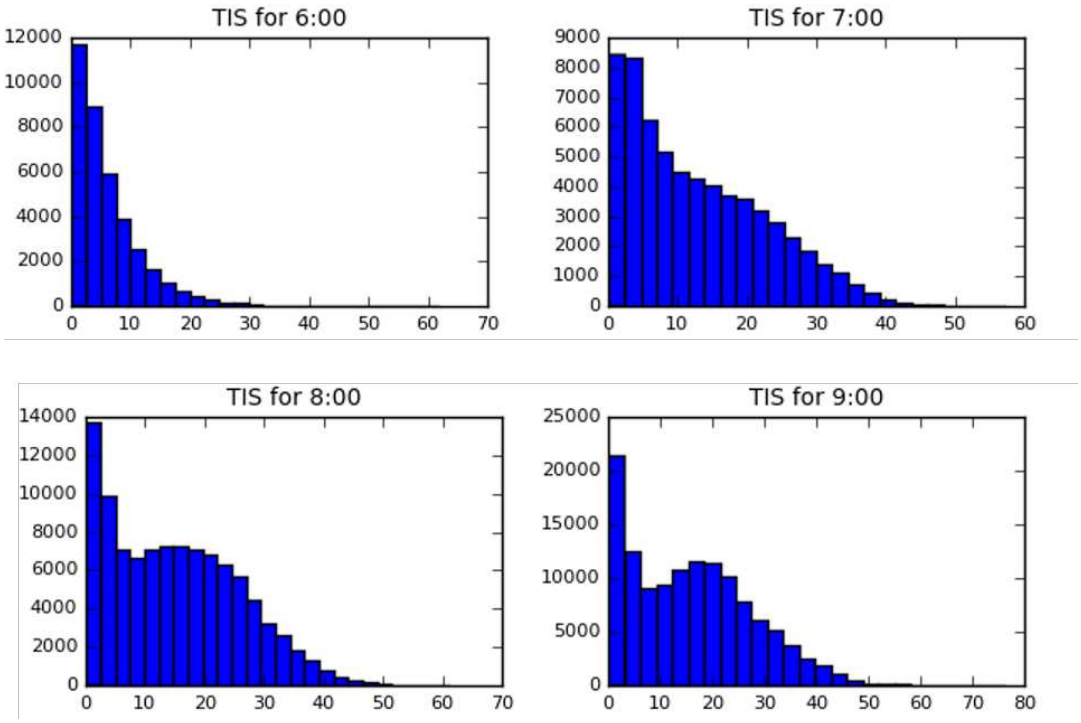


Figure 7. Histograms of individual TIS values for passengers arriving between 6:00 a.m. and 10:00 a.m.

There is clearly additional effort required to compute $\hat{\sigma}^2$, so it is worth asking whether it is different from simply computing the sample variance of all of the observed waiting times

$$S^2_{\text{all}} = \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{M_j}(Y_{ij} - \bar{Y})^2$$

where $N = \sum_j M_j$. Tedious algebra shows that not only is $S^2_{\text{all}} \neq \hat{\sigma}^2$ in general, but S^2_{all} gives too much weight to waiting times that came from time buckets that had larger numbers of observations, even if such intervals are unlikely.

The initial experimental results are shown in Table 1. The table compares the mean and standard deviation values from the SMORE plots (the standard deviation was computed using the reported confidence interval half-width) with the values computed using the method described above. Note that our results match our intuition – there is no discernable difference in the computed means, but the SMORE plot standard deviation values are significantly lower than their counterparts computed using the appropriate method. This is because the SMORE plot standard deviations are for the bucket means over replications and the computed standard deviations are for individual observations. In the context of our example, the traveler using the SMORE plot to estimate the risk of being late (or early) would underestimate that risk. This is an important result if you do not want to miss your flight – note that the underestimation is severe in the bucket that includes the 7:30 flight.

We now turn our attention to computing quantiles (percentiles) using the time-bucket approach. The quantiles displayed in the SMORE plots are critical components for assessing the "risk" associated with the performance metric (Nelson, 2008). However, since the quantiles in the standard SMORE plots are for the mean values over the entire day or by bucket (as described above), we need a different method for calculating the quantiles for the virtual waiting times. The key to our approach is that we have been

treating our data as coming from a mixture distribution: there is an (outer) distribution of time buckets – characterized by the number of observations in a bucket M_j – and an inner distribution of waiting times Y_{ij} within a bucket.

Table 1: Comparison of the simulation-reported and computed results for means and standard deviations. The deltas are computed so that negative values indicate that the simulation underestimates the corresponding metrics.

Simulation-reported Results			Computed Results			Deltas	
Hr	*Mean*	*Std. Dev.*	*Obs*	*Mean*	*Std. Dev.*	*Mean*	*Std. Dev.*
6:00	0.108	0.057	28299	0.108	0.093	-0.41%	-38.96%
7:00	0.265	0.114	46795	0.265	0.153	0.14%	-25.46%
8:00	0.319	0.136	75413	0.319	0.156	-0.07%	-12.62%
9:00	0.343	0.158	93939	0.343	0.174	0.02%	-8.91%
10:00	0.396	0.197	93835	0.396	0.209	-0.02%	-5.88%
11:00	0.326	0.219	74576	0.326	0.232	-0.03%	-5.72%
12:00	0.239	0.217	65673	0.239	0.228	0.16%	-4.93%
13:00	0.263	0.270	65448	0.263	0.286	-0.04%	-5.64%
14:00	0.533	0.337	56216	0.533	0.359	0.07%	-6.00%
15:00	0.582	0.391	37832	0.582	0.401	-0.05%	-2.47%
16:00	0.422	0.392	28405	0.422	0.405	-0.09%	-3.17%
17:00	0.236	0.375	22251	0.236	0.386	-0.11%	-2.85%
18:00	0.247	0.349	20953	0.247	0.360	0.14%	-3.00%
19:00	0.264	0.322	20639	0.264	0.334	0.09%	-3.53%
20:00	0.277	0.299	20988	0.277	0.313	0.04%	-4.45%
21:00	0.238	0.262	18454	0.238	0.276	0.13%	-4.96%

Usually to get a q quantile (where $0 < q < 1$) estimate we invert an empirical cdf; in this case the empirical cdf is a mixture distribution, and we want to find y_q such that

$$\hat{F}(y_q) = \frac{1}{n}\sum_{j=1}^{n} \hat{F}_j(y_q) = \frac{1}{n}\sum_{j=1}^{n} \frac{1}{M_j} \#\{Y_{ij} \le y_q\} \approx q$$

Now consider the pairs (Y_{ij}, w_j) where $w_j = 1/M_j$. Sort all of the Y_{ij}'s from smallest to largest, but keep the correct weight associated with each one. For notation, we now have $(Y_{(i)}, w_{(i)})$ for $i = 1,2,\dots,N$ where $N = \sum_j M_j$ is the total number of waiting times, and we have sorted the Ys from smallest to largest. We now want to find the smallest value of m such that

$$\sum_{i=1}^{m} w_{(i)} \ge nq$$

Then our quantile estimate is $Y_{(m)}$. As an approximate $(1-\alpha)100\%$ confidence interval for y_q we can use two additional sorted values, $(Y_{(L)}, Y_{U)})$, obtained as follows:

$$\text{Largest } L \text{ such that } \sum_{i=1}^{L} w_{(i)} < nq_L$$

$$\text{Smallest } U \text{ such that } \sum_{i=1}^{U} w_{(i)} \geq nq_U$$

where $q_L = q - z_{1-\alpha/2}\sqrt{q(1-q)/n}$ and $q_U = q + z_{1-\alpha/2}\sqrt{q(1-q)/n}$ (note that we do not show the confidence intervals in the tables below, but the available code computes them). This CI is a generalization of the normal approximation to the standard nonparametric confidence interval for a quantile based on the binomial distribution; see, for instance, Banks *et al*. (2010). This approximation will be best when the number of replications n is large. This estimator will likely be poor when the number of observations in a bucket is small; in that case the sample quantile from the sorted data maybe be better.

Table 2: Comparison of the simulation-reported and computed results for 25th (LP) and 75th (UP) percentile values. The deltas are computed so that negative values indicate that the simulation underestimates the corresponding metrics.

Simulation-reported Results				Computed Results				Deltas	
Hr	*Mean*	*LP*	*UP*	*LP1*	*UP1*	*LP2*	*UP2*	*LP*	*UP*
6:00	0.108	0.066	0.1324	0.037	0.152	0.038	0.157	79.4%	-12.9%
7:00	0.265	0.182	0.3424	0.146	0.369	0.151	0.376	25.0%	-7.2%
8:00	0.319	0.227	0.4048	0.206	0.421	0.210	0.424	10.0%	-3.9%
9:00	0.343	0.222	0.4469	0.216	0.456	0.220	0.459	2.7%	-2.0%
10:00	0.396	0.250	0.5231	0.243	0.531	0.245	0.533	2.7%	-1.5%
11:00	0.326	0.138	0.4566	0.136	0.469	0.138	0.470	1.6%	-2.6%
12:00	0.239	0.073	0.3529	0.063	0.352	0.064	0.353	15.9%	0.3%
13:00	0.263	0.088	0.3360	0.072	0.346	0.074	0.348	22.1%	-2.9%
14:00	0.533	0.297	0.6752	0.276	0.707	0.280	0.713	7.7%	-4.5%
15:00	0.582	0.300	0.7504	0.297	0.760	0.305	0.772	0.9%	-1.3%
16:00	0.422	0.114	0.5855	0.106	0.607	0.109	0.613	7.3%	-3.5%
17:00	0.236	0.057	0.2162	0.032	0.253	0.032	0.256	79.4%	-14.5%
18:00	0.247	0.081	0.2358	0.056	0.270	0.059	0.275	44.8%	-12.7%
19:00	0.264	0.092	0.2975	0.070	0.319	0.074	0.330	31.7%	-6.7%
20:00	0.277	0.100	0.3321	0.078	0.358	0.084	0.373	27.6%	-7.2%
21:00	0.238	0.081	0.2861	0.060	0.308	0.063	0.320	34.7%	-7.1%

Table 2 compares the quantile values from the simulation with those computed as described above *(LP1* and *UP1*). For comparison, we also computed the quantiles by simply sorting all of the individual bucket values (*LP2* and *UP2*). As before, we see a significant underestimation of the risk in the simulation-reported LP/UP values over the computed results LP1/UP1. Specifically, LP overestimates the lower percentiles and UP underestimates the upper percentile, significantly shrinking the "likely region" as described by Nelson (2008). Note that this over/under estimation is more pronounced during the "less busy" times in our example model – again, in line with our intuition. Both estimators are computed from the same set of simulation output data – there is nothing wrong with the simulation itself – our point is that an appropriate estimator matters.

4 CONCLUSIONS

In a very practical sense, virtual waiting time is the performance measure of most interest to customers, and few service systems are actually stationary so time of arrival typically matters. Default simulation output analysis, however, emphasizes system-level performance measures and averaging through time. This paper illustrates that daily waiting-time averages are often substantially different from what individual customers should expect, and the variability of those averages is almost unrelated to the

variability of an individual customer's waiting-time experience. Averages within time buckets are far more relevant to individual customers, but even then care must be taken to correctly assess the variability around this average.

REFERENCES

Banks, J., J. S. Carson, B. L. Nelson, and D. M. Nicol. 2000. *Discrete-event System Simulation*. 5th edition. Prentice Hall, Upper Saddle River, NJ.

Gross, D., J. Shortle, J. M. Thompson, and C. M. Harris. 2008. *Fundamentals of Queueing Theory*. 4th edition, Wiley, NY.

Nelson, B.L. 2008. "The MORE Plot: Displaying Measures of Risk & Error." *Proceedings of the 2008 Winter Simulation Conference*, pp. 413-416.

Shao, J. and D. Tu. 1995. *The Jackknife and Bootstrap*. Springer.

AUTHOR BIOGRAPHIES

JEFFREY S. SMITH is the Joe W. Forehand Professor of Industrial and Systems Engineering at Auburn University. He is a Fellow of IIE and has served as the WSC Business Chair and General Chair and is currently on the WSC Board of Directors. His research focuses on simulation design, applications, and education and he is a co-author of *Simio and Simulation: Modeling, Applications, Analysis*. His email and web addresses are jsmith@auburn.edu and http://jsmith.co.

BARRY L. NELSON is the Walter P. Murphy Professor of the Department of Industrial Engineering and Management Sciences at Northwestern University. He is a Fellow of INFORMS and IIE. His research centers on the design and analysis of computer simulation experiments on models of stochastic systems, and he is the author of *Foundations and Methods of Stochastic Simulation: A First Course*, from Springer. His e-mail and web addresses are nelsonb@northwestern.edu and www.iems.northwestern.edu/˜nelsonb.