QUEUEING THEORY AND PROCESS FLOW PERFORMANCE

Chang-Sun Chin[1]

ABSTRACT

Queuing delay occurs when a number of entities arrive for services at a work station where a server(s) has limited capacity so that the entities must wait until the server becomes available. We see this phenomenon in the physical production environment as well as in the office environment (e.g., document processing). The obvious solution may be to increase the number of servers to increase capacity of the work station, but other options can attain the same level of performance improvement.

The study selects two different projects, investigates their submittal review/approval process and uses queuing theory to determine the major causes of long lead times. Queuing theory provides good categorical indices—variation factor, utilization factor and process time factor—for diagnosing the degree of performance degradation from queuing. By measuring the magnitude of these factors and adjusting their levels using various strategies, we can improve system performance. The study also explains what makes the submittal process of two projects perform differently and suggests options for improving performance in the context of queuing theory.

KEY WORDS

Process time, queueing theory, submittal, variation, utilization

INTRODUCTION

Part of becoming lean is eliminating all waste (or *muda* in Japanese). Waste is "any activity which consumes resources but creates no value" (Womack and Jones 2003). Waiting, one of seven wastes defined by Ohno of Toyota (Ohno 1988), can be seen from two different views: "work waiting" or "worker waiting." "Work waiting" occurs when servers (people or equipment) at work stations are not available when entities (jobs, materials, etc) arrive at the work stations, that is, when the servers are busy and entities wait in queue. "Worker waiting" occurs when servers at work stations are ready to serve, but entities are not available, that is, no jobs arrive at the work stations so servers are idle. However, it is clear that both cases consume resources without creating value; "work waiting" consumes space for entities to wait until the server is ready, and "worker waiting" consumes server's capability without actual production. This study explores the underlying causes of waiting in a process flow and finds improvement methods from the queuing perspective.

[1] Ph.D., Honorary Fellow, Construction Engineering and Management Program, Department of Civil and Environmental Engineering, University of Wisconsin, Madison, chin2@wisc.edu

MEASURING KEY FLOW METRICS

Two projects that are similar in terms of the type of building, project budget, and construction duration were selected for the study. We gathered their actual submittal processing times and measured key flow performance metrics for further investigation. Table 1 provides a summary of project characteristics and their key flow components.

Table 1: Comparison of Submittal Process Performances

Project		Project A	Project B
Type of Building		Laboratory + Hospital	Hospital
Location		Wisconsin, USA	Wisconsin, USA
Budget		$144 mil.	$134 mil.
Construction Duration		38 mo.	40 mo.
Sample Size for the Study		641	486
Key Flow Metrics			
Contractor-Want-Time (CWT)[2] (days)	Average	11.58	9.68
	StdDev	7.55	6.32
	CV[3]	0.65	0.65
Actual Lead Time (ALT) (days)	Average	28.79	39.31
	StdDev	27.48	41.97
	CV	0.95	1.07
Variance-To-Want (VTW)[4] (days)	Average	8.11	11.01
	Min	-159.00	-2.00
	Max	116.00	203.00
Inter-arrival Time[5] (days)	Average	2.72	3.45
	StdDev	2.24	4.76
	CV	0.83	1.38
Batch Size (# of Submittals)	Average	2.048	2.06
	Min	1	1
	Max	9	10
Throughput (TH, submittals/day)	Average	0.70	0.60
Work-In-Progress (WIP) (submittals/day)	Average	21.87	23.72
	Min	1	0
	Max	51	65

CAUSES OF LONG LEAD TIME

Little's Law (Work-In-Process (WIP) =Cycle Time (CT) x Throughput (TH)), which is quite general and applies to any queue discipline, specifies how WIP and flow time in the system are linked (Hopp 2007; Hopp and Spearman 2000; Little 1961). A system containing a large amount of WIP inevitably results in long lead times or, conversely, a system with reduced WIP has faster responses (Hopp 2007; Hopp and Spearman 2000; Lambrecht and Vandaele 1994). However, it is possible to have two different conditions with the same throughput (i.e., TH=WIP/CT), i.e., either a long cycle time and large WIP or a short cycle time and small WIP.

[2] The response time expected by the contractor

[3] Coefficient of Variation

[4] The difference between CWT and ALT, calculated from the difference between CWT and ALT; Following the usual convention, early, on-time, and late responses will have negative (-), zero (0), and positive (+) values, respectively.

[5] Inter-arrival times are simply the times between the arrivals of entities to the process.

In the selected cases, we observed that the THs of the two projects are only slightly different at 0.7 and 0.6 submittals/day, respectively. As for other parameters, the WIP level of Project A (21.87 submittals) is lower than that of Project B (23.72 submittals), and average actual lead times of Project A (28.79 days) are much shorter than those of Project B (39.38 days). Of course, any manager would prefer the system with low WIP and short cycle times (like Project A) since such a system is more efficient in the sense of its ability to convert WIP into throughput (Hopp 2007). The VTWs of Projects A and B are 8 days and 11 days, respectively, which result can be interpreted to mean that, on average, returning submittals to the contractor in Project A takes three days less than it does with Project B. This is additional evidence that Project A is more efficient than Project B. What makes the performances of the two projects different?

QUEUING DELAY

There are several causes of delay. One of the most important is queuing delay when a large proportion of flow time is spent waiting in queue (Hopp and Spearman 2000). Since the 1990s, Just-In-Time (aka, Lean Production or Toyota Production System), Time-based Competition and Fast Cycle Time strategies have given rise to a renewed interest in queuing (Lambrecht and Vandaele 1994). Researchers in lean construction also have gained insights into queuing theory (Bertelsen et al. 2007; Koskela 2004). Any fast cycle time strategy deals with reduced waiting times, a fact well documented through the queuing theory, the study of waiting-line phenomena (Hopp 2007).

Queuing delay occurs when a number of physical entities arrive for service at a server or servers that have limited capacity, and the entities must wait until a server becomes available (Hopp and Spearman 2000; Lambrecht and Vandaele 1994). One of the well established general types of queuing models, where the arrival time and service time can take on any probability distribution, is the G/G/1 model (Hopp 2006). The first G denotes the distribution of inter-arrival times, the second G denotes the distribution of effective process times, and the number 1 describes the number of servers at the workstation.

Equation 1: G/G/1 Queueing Equation

$$W_q^{G/G/1} \approx \left(\frac{c_a^{\,2} + c_s^{\,2}}{2} \right) \left(\frac{\rho}{1-\rho} \right) \tau = V \text{ x } U \text{ x } T$$

The equation is also known as the VUT equation or Kingman's equation, named after one of the first queuing researchers to propose it. In the equation, C_a, C_s, ρ, and τ denote the inter-arrival time's Coefficient of Variation[6] (CV), the process time CV, utilization, and average service time, respectively. The expression gives that queuing delay consists of the multiples of variation in the inter-arrival time, effective process

[6] Coefficient of variation is the ratio of the standard deviation to the mean and is unitless because the mean and standard deviation have the same units. Hopp and Spearman (2000) established the classification of this variation into low, moderate and high variation based on the magnitude of the CV value: LV (low variation) for CVs less than 0.75, MV (moderate variation) for CVs between 0.75 and 1.33, and HV (high variation) for CVs greater than 1.33.

time, utilization and average service (process) time. Therefore, total process time of a system experiencing queuing delay is the sum of the queuing delay and process time. By exploring the underlying causes of these parameters in the VUT equation, we can simply but systematically identify the factors that cause waiting in a given queuing system (Hopp 2007; Hopp and Spearman 2000; Hopp et al. 1990).

VARIABILITY (V) FACTOR

As shown in the queuing delay equation, cycle time is related to the average of flow time (T factor) as well as to the variations of flow time (V factor) and the utilization of server at workstation (U factor). Hence, even if the process time were stable (i.e., the process time's CV is low), the waiting time in the queue will increase because of inter-arrival variations that result in an increase of the V factor in the queuing equation. In order to explain the variation effect, Figure 1 illustrates two different entity arrival patterns. One pattern is that of a low-variation-arrival process and the other is that of a high-variation-arrival process. The low variation arrivals are smooth and regular, while the high variation arrivals are "bursty" and uneven (Hopp 2007). Hence, any efforts to make intervals smoother and more regular will improve the system performance.

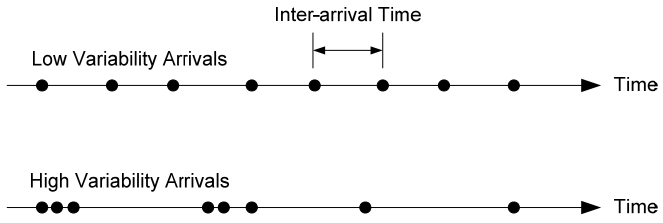

Figure 1: High and Low Variation Arrivals (Hopp 2007)

Inter-arrival time variation. The entity arrival patterns of each project are different. The inter-arrival time of Project A (CV=0.83) is less variable than that of project B (CV=1.38).

Submittals are not usually sent to the designer one at a time, but are more often batched together with different expected response times. The batch sizes of Projects A and B have almost the same profiles, i.e., on average, about two (2) submittals created and sent to the design team on the same date for review. However, batching is another cause of increased inter-arrival variation (Hopp 2007; Hopp and Spearman 2000). One might think that the variation with batching is zero because entities that are batched arrive at a workstation simultaneously. However, if we look at the inter-arrival times of each entity in the batch from the perspective of the individual submittal, we see a different picture (Hopp and Spearman 2000). For example, in Project A, an average of 2.05 submittals are batched and delivered to the reviewer at the same time. However, the reviewers can do only one review at a time. The inter-arrival time (i.e., the time between the arrival of the current submittal and that of the previous one) for the first submittal in the batch is 2.72 days (for average inter-arrival time, see Table 1), and zero only for the next 1.05 submittals (2.05-1). Hence, the mean time between arrivals is 1.33 days (2.72 days divided by 2.05 submittals), and the variation of these times would be:

$$\sigma_a^2 = \left[\frac{1}{2.05}(2.72)^2 + \frac{1.05}{2.05}(0)^2\right] - t_a^2 = \frac{1}{2.05}(2.72)^2 - (1.33)^2 = 1.84$$

Therefore, the arrival Squared Coefficient of Variation (SCV) is:

$$C_a^2 = \frac{1.84}{(1.33)^2} = 1.04$$

Because of the batch arrival, the batch arrival's CV is higher ($\sqrt{1.04}$ = 1.02) than the inter-arrival time's CV (0.83) or the process time's CV (0.95); thus, the batching effect increases the flow variation significantly and degrades the system performance, resulting in longer cycle time. If we calculated the CV that is a result of the batching effect of Project B, we will get C_a = 1.07, which is slightly higher than that of Project A (1.02). Thus, the variation effects of the two projects resulting from batch arrival are almost equal.

Process time variation. Process time patterns also can be explained using the illustration in Figure 1. As the figure shows, the processing time of Project A (0.95) is less variable than that of project B (1.07). Hence, we can conclude that Project A is a more stable system than Project B because it has less variability in its inter-arrival rate, processing time and batching. However, we will also need to look at other factors (U and T factors) to ensure that Project A is more efficient than Project B.

UTILIZATION (U) FACTOR

Utilization is the fraction of time a workstation is not idle for lack of parts. Utilization is computed as:

Utilization (ρ) = Entity Rate into workstation / Capacity of workstations (Hopp and Spearman 2000),

where entity rate into workstation is equivalent to the entity inter-arrival rate and capacity of a system is the maximum average rate at which entities can flow through the system. Relating to the queuing equation (Equation 1), the utilization factor (U) will be proportional to ρ/(1-ρ), where ρ is the station utilization (Hopp 2007). Hence, in theory, as station utilization reaches 100% (i.e., 1), queuing delay would approach infinity. If the entity arrival rate into the workstation exceeds the capacity of the workstation, waiting will begin because the workstation is not capable of processing entities that flow in. For example, if three jobs arrive at workstation every hour, but the system can process only two jobs per hour, utilization of the system would be 100%, not 150% since utilization cannot exceed 100%. In this system, one job should be in queue until the workstation is available. However, if three jobs arrive at workstation every hour, but the system can process four jobs per hour, utilization of the system would be 75% and no queuing delay would occur because the system is fully capable of processing jobs without delay.

"Worker waiting" can also be explained by queuing theory. When a worker is idle, no entities arrive at the workstation, so the entity rate into the workstation is 0, resulting in 0% station utilization (ρ). Hence, the U factor (ρ/(1-ρ)) becomes 0, resulting in no queuing delay and a total process time of 0.

In project A, an average of 2.048 submittals arrives at the review system every 2.72 days. Hence, the average entity rate into the workstation would be 0.75 submittals per day (2.048 submittals per 2.72 days). The capacity is equivalent to the

throughput rate if the system reaches its maximum capacity, so it is clear that Project A is incapable of keeping up with the arrival rate of submittals and the WIP will build up over time to an average of 22 submittals and a maximum of 51 submittals. The utilization of the review system of Project A would reach 100% so, in theory, WIP can explode infinitely. Project B will have 100% utilization for the same reason.

PROCESS (T) TIME FACTOR

The two systems show huge differences in their average process times; the average process time of Project A (28.79 days) is much shorter than that of Project B (39.31 days). In the queuing equation, processing time is the average ***effective*** process time for an entity at the station and is measured as the time from when an entity reaches the head of the line to when it is finished (Hopp 2007). Under this definition, effective process times include such detractors as machine failures, setup times, operator breaks, or anything else that extends the time required to complete processing of the entity (Hopp 2007; Hopp and Spearman 2000). Figure 2 shows the detractors in the effective process time. Minimizing or eliminating detractors will shorten the process time by increasing the proportion of pure execution time within the effective time.

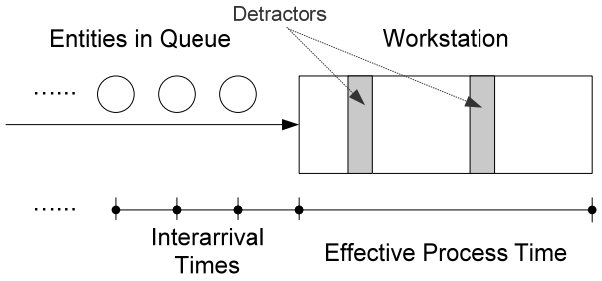

Figure 2: Effective Process Time and Detractors In A Line Flow

INCREASING PROCESS PERFORMANCE

In queuing theory, lead time is affected by variation (V factor), utilization (U factor) and process time (T factor). We can attain a shorter process lead time by investigating the root causes that degrade each factor.

REDUCING THE VARIATION FACTOR

Two variation components that cause waiting in line have been identified as inter-arrival time and process time variations. Directing an improvement effort toward making these variation components more consistent would narrow the span and lower the variation. Among the many techniques dealing with variation, a load-leveling technique is used to alter the distribution of arrival times, and standard setup alters the distribution of process times (Muir 2006). Hopp and Spearman (2000) states that high variability will be most damaging at work stations with high utilization because such variations will create a bottleneck. Another option for flow variation reduction includes reducing batch sizes as discussed previously.

REDUCING THE UTILIZATION FACTOR

In the queuing equation, in theory, as utilization approaches 100%, any variation in inter-arrival times and process times can drive wait time to infinity, so merely increasing utilization (rate-in/capacity of a workstation) to make up for lost progress can cause an increase in waiting time (the largest portion of the flow time) unless variation is reduced. Consider one extreme case whose variability factor (V) is 2. If the process time (T) is 1 hour and the utilization (ρ) is 50%, the queuing delay would be 2 hours (VUT = 2 x 1 x 1), resulting in total process time of 3 hours (total process time = queuing delay + process time = 2 + 1). Therefore, it is common that the larger the variation in the processing time, the more safety time (as a buffer) is needed to absorb the impact of the variation. However, independently considering the variation effect, two methods can reduce utilization: reduce entity rate-in or increase the capacity of the workstation.

The only way to reduce entity rate-in is to reduce the number of entities arriving at a workstation. Submittals are generally created by subcontractors who use a master schedule to plan their submittal package production times and refer to a submittal schedule prepared by a general contractor. Hence, reducing the number of submittals can be achieved by demand control, load-leveling or batch-size reduction. Theoretically applicable techniques to achieve this goal include Kanban (pull), Heijunka and One-piece flow of the Toyota Production System (Hopp and Spearman 2000; Muir 2006).

Any efforts directions to increase the capacity of a work station—assigning more reviewers, training reviewers to improve their skills, increasing reviewer's available time—will reduce the utilization of a system.

REDUCING PROCESS TIME

Process time can be reduced by either direct or indirect methods. The direct method increases capacity by increasing the number of servers or workstations, such as by assigning more engineers to the submittal review process. The indirect methods include standardization, automation, training workers, and minimizing/eliminating detractors to increase the proportion of pure process (execution) time within the effective process time.

SUMMARY OF POSSIBLE CAUSES AND IMPROVEMENT METHODS

As a final step, the author conducted a series of brainstorming sessions and interviews with a design team (reviewers) in order to identify the possible causes corresponding to each factor and to find available improvement methods. Table 2 summarizes the VUT factors, possible causes, and improvement methods for the selected submittal process, based on the queuing theory. First, for the V factor, two directions can be considered: reducing inter-arrival time and reducing process time variation. Any efforts to narrow the intervals between arrivals or those between the processing start and end times will reduce the flow variation.

For the U factor, expanding capacity in order to lower utilization is generally costly, so maintaining utilization as high as possible is usually desirable. The only way to keep high utilization without increasing waiting time is to have a low

variability factor. For this reason, variability reduction is often the key to achieving highly efficient systems (Hopp 2006; Hopp 2007; Hopp and Spearman 2000).

For the T factor, improvement should be directed to either increasing the servers' capacity or to improving the quality of submittals and design documents.

Table 2: Possible Causes and Improvement Methods

VUT Factors		Possible causes	Improvement methods
Variation factor $\left(\frac{c_a^2 + c_s^2}{2}\right)$	Inter-arrival time variation (C_a)	Contractor's decision	Reduce the chances of abusing submittals.
		Scheduling	Generate demand forecasts and well prepared submittal schedules.
		Transmittal delays	Adopt electronic data interchange method.
		Variety of submittal	Use standardized format.
		Upstream processing	Reduce variation in upstream processes (i.e., variation in submittal package preparation).
	Process time variation (C_s)	Reviewer availability	Reduce detractors and create flexible working hours.
		Reviewing speed and reviewer's skill level	Standardize the review process and train reviewers.
		Document quality	Use standardized format.
		Project information quality	Improve the quality of project information (specifications and drawings).
		Task variety	Categorize or classify tasks based on key priorities.
Utilization factor $\left(\frac{\rho}{1-\rho}\right)$	Utilization (ρ) = Rate-in /Capacity	Entity arrival rate into workstation	Control arrival rate by schedule and demand control and reduce batch sizes.
		Number of reviewers	Adding resources (equipment or staff).
		Reviewer availability	Reduce the detractors.
		Reviewing speed and reviewer's skill level	Train reviewers, assign a designated knowledgeable individual, use flexible labor, etc.
Process time (τ)		Reviewer availability	Reduce detractors and create flexible working hours.
		Reviewing speed and reviewer's level of expertise	Reduce the process time by means of standardization, automation or training.
		Document quality	Use standardized format.
		Project information quality	Improve the quality of project information (specifications and drawings).
		Task variety	Categorize or classify tasks based on key priorities.

CONCLUSION

A system with a queuing delay is affected by variability, utilization and process time. Various options are available to reduce the flow time and improve the flow performance, but improvement would be simpler and clearer with greater understanding of queuing theory. Evaluating each factor (V, U, and T) helps to identify possible problem areas and design more tailored improvement strategies in order to yield the best results while minimizing unnecessary effort and undesirable effects.

REFERENCES

Bertelsen, S., Henrich, G., Koskela, L., and Rooke, J. (2007). "Construction Physics." Proc. of the 15th Annual Conference of the International Group for Lean Construction (IGLC-15), July 18-20, Michigan, USA.

Hopp, W. (2006). Single Server Queueing Models, to appear in When Intuition Fails: Insights from Simple Models, Dilip Chhajed, Tim Lowe (eds.), Springer, New York, Available at http://webuser.bus.umich.edu/whopp/publish.htm, Visited on Feb 18, 2009.

Hopp, W. (2007). Supply Chain Science, McGraw-Hill/Irwin.

Hopp, W. J., and Spearman, M. L. (2000). Factory Physics: Foundations of Manufacturing Management, Irwin/McGraw-Hill, Boston, MA.

Hopp, W. J., Spearman, M. L., and Woodruff, D. L. (1990). "Practical Strategies for Lead Time Reduction, American Society of Mechanical Engineers." Manufacturing Review, 3(2), 78-84.

Koskela, L. (2004). "Making Do - The Eighth Category of Waste " Proc. 12th Annual Conference of the International Group for Lean Construction (IGLC-12), August 3-5, Denmark.

Lambrecht, M., and Vandaele, N. (1994). "Queueing Theory and Operations Management." Tijdschrift voor Economie en Management, XXXIX(4).

Little, J. D. C. (1961). "A Proof for the Queuing Formula: L=λW." Operations Research, 9, 383-387.

Muir, A. (2006). Lean Production Six Sigma Statistics-Calculating Process Efficiency in Transactional Projects, McGraw-Hill, New York, NY.

Ohno, T. (1988). Toyota Production System: Beyond Large-scale Production, Productivity Press, Portland, OR.

Womack, J. O., and Jones, D. T. (2003). Lean Thinking: Banish Waste and Create Wealth in Your Corporation, Free Press, New York, NY.