Answers to More Questions from Webinar Attendees

Webcast: "Balance and Bottlenecks and Triggers, Oh My!"

Presenter: Dr. Lars Maaseidvaag, Senior Master Black Belt, MoreSteam.com

September 28, 2012

Q: What are the differences between Flow Time and makespan and throughput time?

These terms appear to be quite similar, as they all refer to the amount of time it will take to get from a defined starting point to a defined ending point in a process. Looking at some online references, it looks like Flow Time may be more commonly used for service environments, while Makespan and Throughput have several manufacturing references.

One reference on Makespan indicated the term can also refer to the time to make several different products in a process. In this context, Makespan would also provide an indication of the flexibility of the process in terms of how quickly the process can cycle through all of the items it produces.

Regardless of the terms used, the concepts are straightforward. These various terms are concerned with the total time through a process, including all of the various non-value added waste, value-add, queue time, etc., or simply the lead time through the process. Makespan potentially adds another dimension, which is the interaction of all parts or items in the process. If setup time is high and therefore batch sizes are high, then the process becomes inflexible to short term changes in customer demand patterns. It's hard to change to making red widgets if our process is stuffed full of blue, gold, and green widgets. Whatever the term, excess inventory in a process leads to a long delay to flush the current WIP from the process in order to introduce new options.

Q: How were the P-values calculated? Do we have a Z value in this numbers?

The Z value would be the takt time of 7. The Excel function "=NORMDIST(7,4,2,TRUE)" gives you the probability that the task time would be less than or equal to the takt time. For this example, 7 is the 'x' value in Excel's terminology (our Z value), 4 is the mean, 2 is the standard deviation, and the 'TRUE' indicates that the function will return the cumulative probability of the distribution rather than the probability mass function (which is just the height of the curve at the x value of 7).

"Q: Since you are dealing with cycle times, the data at the lowest level would be skewed or non-normal. Any practical suggestions on how to sub-group the data to move the data toward normality?

Process Capability is quite sensitive to the normality constraint, and the skew in cycle time data is almost always to the right or upper specification limit. As such, right skewed data will tend to mean the Cpu calculation will be overly optimistic and underestimate the probability represented in the right tail of the distribution. However, in general the assumption of independence between processing times in the steps and the arrival of demand is probably a bigger problem in most cases. Using Cpu will be directionally correct and will help point you towards the problem steps in your process. As in all areas of continuous improvement, strict math must join experience and judgment in the analysis process.

If your processing times in each step are truly independent, and demand arrivals are truly independent, then the Cpu and the rolled throughput yield calculation I showed are pessimistic estimates. In this case, you may be fine with some degree of non-normality. However, it will be a judgment call you will have to make based on your experience, and if you need to, simulate the process with a discrete event simulator and then you can model the process directly with the non-normal data.

MoreSteam's Senior Statistician, Smita Skrivanek, adds:

On the issue of non-normality, my take is: It is true that most process data, especially cycle time, tends to be skewed right, and the commonly used process capability formulas require normality, so I agree with your analysis that the Cpu will tend to underestimate the percent nonconforming.

If the non-normality is marked, I generally recommend a suitable normalizing transformation be applied to the data. Logs and square-roots tend to work well for right-skewed, non-zero, positive data (cycle times). This usually corrects the problem enough that the capability indices are more reliable.

Having said that, there are ways to estimate capability indices for non-normal data by fitting the best-fit distribution (usually lognormal or weibull) and using the percentiles from the fitted distribution to estimate capability.

Note: Minitab does non-normal capability analysis for various distributions (process still must be in control.)

Q: Depending on the amount of variation, the average is hardly ever the "normal" value - it may not even be the most frequent value (that's the mode).

True, 'average' rarely happens. In simulation, we often use the triangular distribution to represent business processes. The triangular distribution is appealing because it has a defined minimum (equal to the pure value-added time of the process), and distinct mode, and a defined maximum (usually defined as the point where there is an intervention to complete the work and the process effectively changes to a new process). The Erlang distribution is also widely used as it provides a distinct mode, no negative values, and skew to the right tail of the distribution.

Q: What is the meaning of Cpu? Is it a probability? If so, it is the probability for what?

Cpu is part of the Six Sigma Process Capability toolset. Cp and Cpk refer to the capability of a process to meet user specifications. A Cp of 1.0 means that plus and minus 3 standard deviations of variation in the process output are still within the customers upper and lower specification limits. A Cp of 2.0 is a “Six Sigma” process, and virtually all process variation still falls within the customer specification limits.

For the Takt time application of Process Capability, we are not concerned with task times which are shorter than the average as the task just completes early. We are only concerned with the task times which run long. If they run too long, they will delay the process and cause problems. This upper specification limit on task time is the takt time or rate of customer demand. Cpu measures just the upper tail of the process distribution. So, finally, we can say that a Cpu of 1.0 would indicate that 99.865% of all task times would be within the upper specification (takt time) limit. A Cpu of 2.0 would yield 99.999% within spec.

Q: Should use the actual distributions of the task times (histograms), not means and standard deviations and assume (erroneously) that they are normally distributed

The use of actual data is great (and the best) when you have a large data set to draw from and you can guarantee that there were no confounding factors at work during the time period that the data was collected. Given that those conditions are met, I agree that pulling randomly from a large data set is a very effective method. But it does need to be a large, fully representative data set.

No statistical model is perfect, from the simple calculation of the mean to a complex multiple regression. They are all wrong to some degree, but many are useful (recall George Box’s quote “All models are wrong, some are useful”). The simple and quick application of the Process Capability tool to takt time analysis is imperfect, but arguably less imperfect than a traditional takt time analysis performed just at the averages.

Simulation with a large actual data set is best, but is it worth the effort and cost to make an informed decision of where to apply continuous improvement efforts? That is where experience and your specific process conditions will have to enter the decision making process.