To Monte Carlo or Not to Monte Carlo…. That is the Question!

Authored by

Joe Bauer

PRICE Systems, LLC.

For Presentation at the

ICEAA Professional Development

and Training Workshop

June 12-15, 2018

Phoenix, Arizona

Table of Contents

Background

During cost estimation training events and consulting efforts, the author is often asked about the similarities and differences between two common risk methodologies: Method of Moments and Monte Carlo. These methodologies are used in a number of commercial software tools, such as TruePlanning®, Crystal Ball®, and @Risk®. Cursory observations show that the results are similar. However, a full study of the similarities and differences has not been done with regard to the behavior of these two risk methodologies within a commercial parametric estimating framework such as TruePlanning®.

Independent of the method used, one of the most abstract tasks facing the cost estimator is how to spread risk dollars across a program once the risk analysis has been completed. Programs differ, phases differ, inherent program risks differ; there is no one perfect solution for every situation. The author will examine several risk phasing methodologies to assist the estimator in choosing an appropriate solution.

Monte Carlo Overview

First, consider the risk analysis methodology known as Monte Carlo simulation. Monte Carlo simulation grew out of an attempt to evaluate games of chance in the 1940s.[1] Monte Carlo simulation utilizes randomness (stochasticity) to solve a deterministic problem. It is clearly impossible to execute a program one thousand times to collect a range of outcomes. Simulation is faster, smarter, and of course, cheaper.[2]

Take the case of calculating a software estimate using the code size, productivity (expressed as hours per line of code), and labor rate shown below:

ESLOC * Productivity * Labor Rate = Program Cost

10K * 2.3 * $200 = $4,600,000

This is a deterministic problem, easily done with mental math or a standard calculator. In reality, most acquisition professionals understand that each of these variables likely have some range of possible values in addition to the value stated above. What if the source lines of code are more than predicted? What if the productivity of the development team differs between potential bidders? What if the labor rates differ as well? All of these points are valid and must be dealt with to ensure the program manager has the right amount of funding at the right time. This not only contributes to the success of the program in question. It also contributes to the affordability of an entire portfolio.

The Monte Carlo process provides a way to establish ranges of input values, each value within that range having a probability based on some underlying distribution. Taking the earlier example, we can see how applying ranges around the inputs would lead to a range of possible outcomes.

The Monte Carlo process is as follows: random input values are selected from each input distribution. The value chosen is dependent upon the probabilities associated with the distribution as well as any correlation between input variables. The cost estimating relationship is then calculated with the random input values. The outcome is stored. This process is repeated hundreds or thousands of times. As the outcomes are evaluated, a probability density function is created showing the range of outcomes and frequency within each bin.

The utility is clear when we need to answer acquisition related questions such as: How confident are we in the point estimate? How much risk (dollars) should be applied to a program to achieve a 60% chance of success? If our budget is x dollars, what is the probability of overrunning that amount? Inferential statistics allows the estimator to use the statistical properties of the probability density function and cumulative distribution function to answer all of these questions. Modern software applications such as Crystal Ball® and @Risk® make the process easy to execute and easy to interpret the results.

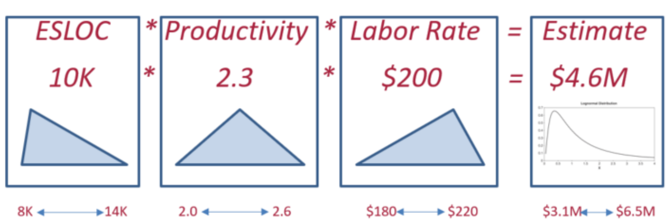
Method of Moments Overview

In contrast to Monte Carlo simulation, Method of Moments offers an alternative method to conducting risk and uncertainty analysis. Method of Moments is simple, fast, and offers similar statistical output that is often generated by popular Monte Carlo simulation tools.

First, the definition of a moment is necessary. Moments are characteristics of a distribution. The first moment is the mean, or average value in a data set. Next, we have the variance, or spread of the underlying data. Finally, there is skewness and kurtosis, which are indicators of distribution symmetry and "peakedness." Peakedness refers to how tall or flat a distribution is in relation to its width.[3]

With Method of Moments, inferences can be made regarding an output distribution based on the mathematical evaluation of one or more input distributions. The simplicity of math, particularly for computers, makes Method of Moments much faster than Monte Carlo.

Like Monte Carlo, ranges around input values are still necessary. However, the use of those ranges differs. In the figure below, we have a similar arrangement as shown previously.



There is a major difference in the way the input ranges are used, however. The pessimistic values for all inputs are combined for a pessimistic or high calculation of the cost estimating relationship. The most likely or point estimate inputs are combined for a most likely calculation. Finally, the optimistic or low values are used to calculate a low value. From these three results (high, mid, low), the mean and standard deviation of <u>each cost element</u> is calculated using the formulas shown below:

$$\mu = \frac{L + M + H}{3} \qquad \sigma = \sqrt{\frac{L^2 + M^2 + H^2 - LM - LH - MH}{18}}$$

There is no random selection from each input range. The highest, lowest, and most likely values are used in combination to generate a highest, lowest, and most likely outcome. As multiple cost elements are combined or rolled up in Method of Moments, the means and standard deviations are determined using the formulas noted below:[4]

$$\mu_s = \sum_{j=1}^{n} \mu_j \qquad\qquad \sigma_s = \sqrt{\sum_{k=1}^{n} \sigma_k^2 + 2\sum_{k=2}^{n}\sum_{j=1}^{k-1} \rho_{jk}\sigma_j\sigma_k}$$

The resultant standard deviation for one cost element is a key cause of differing results between Method of Moments and Monte Carlo simulation. This topic will be explored in great detail shortly after the results are presented using a common case study.

Case Study Introduction

Establishing common ground rules and estimating parameters is important for comparing the two risk / uncertainty methodologies. In this presentation, the author generated a simple point estimate using commercially available parametric estimating software called TruePlanning®. TruePlanning® is an estimating framework and provides the flexibility of employing similar or disparate models within a common framework, thereby also estimating the cost and effort of integration, assembly and test of those cost models.

The scenario used in this evaluation was a development and production effort. The quantity of prototypes was 10 units. There were 100 production units. The hardware was a twenty-pound component with weight evenly distributed between electronics and structure. The software component was 15,000 lines of C code. Effort to integrate, assemble and test the hardware and software was also included. All other inputs in the TruePlanning® framework and cost models were left at default values.

Since the same product breakdown structure and inputs were used for both methods, the same number of cost elements and same cost estimating relationships were also used. Uncertainty distributions within the Method of Moments application were limited to triangular distributions. As such, triangular distributions were also used in the Monte Carlo (Crystal Ball®) analysis. For both cases, key cost drivers were varied plus twenty percent and minus ten percent from the point estimate values.

Many factors could be compared to evaluate the two methodologies. To narrow the scope, the author evaluated the means, standard deviations, 50th percentile and 80th percentile of each method. Each of these indicators were evaluated at various levels of correlation: 0.0, 0.2, 0.5, and 0.7.

Results

The results for each method are first explored, then compared to each other. Next, the magnitude and causes of any differences between the two methods will be discussed. In the tables below, the raw results are presented for completeness.

	Multivariate Parametric Model Three cost elements Correlation at 0.0			
	MoM	**MC**	**Raw Delta**	**% Delta**
Mean	$7,756,171	$7,200,919	$555,252	7.2%
SD	$ 837,878	$ 581,138	$256,740	30.6%
50%	$7,711,306	$7,109,153	$602,153	7.8%
80%	$8,443,035	$7,667,590	$775,445	9.2%
CV	10.8%	8.1%	2.7%	25.3%

	Multivariate Parametric Model Three cost elements Correlation at 0.2			
	MoM	**MC**	**Raw Delta**	**% Delta**
Mean	$7,756,171	$7,169,694	$586,477	7.6%
SD	$ 904,672	$ 620,266	$284,405	31.4%
50%	$7,703,943	$7,080,520	$623,423	8.1%
80%	$8,495,754	$7,661,599	$834,154	9.8%
CV	11.7%	8.7%	3.0%	25.8%

	Multivariate Parametric Model Three cost elements Correlation at 0.5			
	MoM	**MC**	**Raw Delta**	**% Delta**
Mean	$7,756,171	$7,201,829	$554,342	7.1%
SD	$ 996,503	$ 738,776	$257,726	25.9%
50%	$7,692,938	$7,076,426	$616,511	8.0%
80%	$8,567,626	$7,814,235	$753,391	8.8%
CV	12.8%	10.3%	2.6%	20.2%

	Multivariate Parametric Model One cost element Correlation at 0.7			
	MoM	**MC**	**Raw Delta**	**% Delta**
Mean	$7,756,171	$7,235,478	$520,692	6.7%
SD	$1,053,285	$ 849,101	$204,184	19.4%
50%	$7,685,627	$7,066,739	$618,888	8.1%
80%	$8,611,701	$7,910,361	$701,340	8.1%
CV	13.6%	11.7%	1.8%	13.6%

Method of Moments Results

For Method of Moments specifically, the results have been consolidated and presented below:

	Method of Moments			
	0.0	**0.2**	**0.5**	**0.7**
Mean	$7,756,171	$7,756,171	$7,756,171	$7,756,171
SD	$ 837,878	$ 904,672	$ 996,503	$1,053,285
50%	$7,711,306	$7,703,943	$7,692,938	$7,685,627
80%	$8,443,035	$8,495,754	$8,567,626	$8,611,701
CV	10.8%	11.7%	12.8%	13.6%

The correlation for each model run is shown across the columns. The mean, regardless of correlation, remains exactly the same. Recall the equation for the mean of one cost element or a series of cost elements in Method of Moments. The mean value is solely dependent upon the high, mid and low costs. The standard deviation increases steadily as correlation increases. This makes sense, as the correlation value is part of the standard deviation equation in Method of Moments. The 50th percentile shows a slight decrease while the 80th percentile shows a steady increase.

Before the discussion on why the 50th and 80th percentiles behave in this manner, consider how Method of Moments calculates the value for any percentile. The percentile values are determined in part by two variables called P and Q. In turn, P and Q are determined by the mean and variance. See formulas below:[5]

$$P = \frac{1}{2} \ln \frac{\mu^4}{\mu^2 + \sigma^2} \qquad Q = \sqrt{\ln\left(1 + \frac{\sigma^2}{\mu^2}\right)}$$

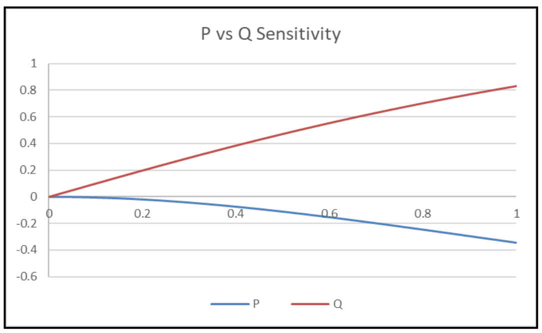
From the equations, it is easy to see that as the standard deviation increases (sigma), so does the variance (sigma squared). As the variance increases, P will decrease since the variance is in the denominator. For Q, the opposite happens since the variance is in the numerator.

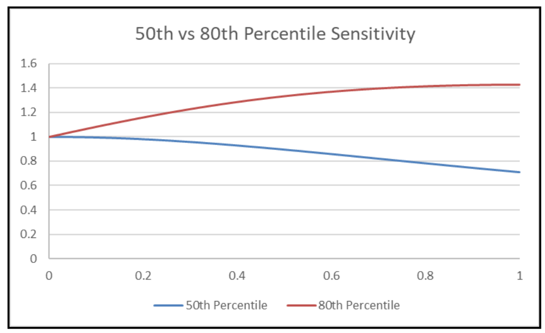
The percentile values are determined by exponentiating P plus Q times Zeta, as shown below. Zeta values for any percentile are available in the back of standard statistics text books. The Zeta value for the 50th percentile is 0.0. The value for the 80th percentile is 0.84162.[6]

$$e^{P+Z_{\alpha}Q}$$

At the 50th percentile, the second term in the exponentiation is zero, so the 50th percentile value is solely dependent upon exponentiating P. Since P decreases as the variance increases (which is caused by an increase in correlation), we see a slight yet consistent decrease in the 50th percentile value as correlation increases.

At the 80th percentile, the combination of P and Q result in an increase in the 80th percentile value as correlation increases. See the charts below for P and Q sensitivity at various correlation levels, as well as how the resultant 50th and 80th percentile values behave at various correlation levels.





Monte Carlo Results

For Monte Carlo specifically, the results have been consolidated and presented below:

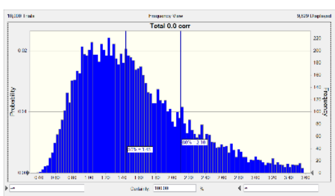
	Monte Carlo			
	0.0	**0.2**	**0.5**	**0.7**
Mean	$7,200,919	$7,169,694	$7,201,829	$7,235,478
SD	$ 581,138	$ 620,266	$ 738,776	$ 849,101
50%	$7,109,153	$7,080,520	$7,076,426	$7,066,739
80%	$7,667,590	$7,661,599	$7,814,235	$7,910,361
CV	8.1%	8.7%	10.3%	11.7%

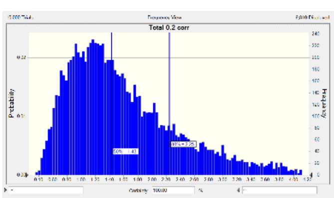
The correlation for each model run is shown across the columns. As expected, the means are variable due to the variable nature of input value selection. The standard deviation and 80th percentile increase as the 50^{th} percentile decreases (as correlation increases). In general, the trends and behaviors of key metrics are similar between the two methodologies, albeit for different reasons.

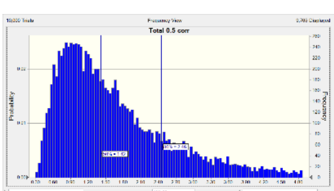
In a Monte Carlo simulation, as inputs are more highly correlated, the randomness of input value selection is modified. Consider two highly correlated inputs. If a random selection is made in one tail of the first input, the high correlation will influence the area of the distribution where the random value is selected for the second input. As a result, there will be more cases where high input values across correlated inputs are chosen for a given model iteration, and more cases where low input values across correlated inputs are chose for a given model iteration run. With “higher highs” and “lower lows” included in the same simulation, the effect is a wider spread of the outcome distribution, which can be identified by a higher standard deviation as correlation increases.

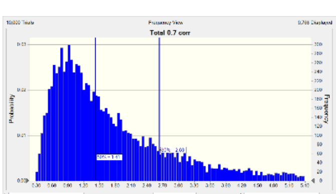
The 50^{th} and 80^{th} percentiles react the same way in Monte Carlo simulation as in Method of Moments. Note: this effect can be amplified or minimized by the cost estimating relationship. Higher order power or exponential equations amplify the effect. In this case study, higher order equations were used.

As correlation increases, the distribution is stretched. Again, this is due to more high and low outcomes included in each simulation. The distribution is not stretched equilaterally. The “higher highs” are greater in magnitude than the “lower lows.” More of the iterations are in the right skewed tail of the outcome distribution, which moves the 50^{th} percentile lower and the 80^{th} percentile higher. Note the graphs below showing the skewness impact due to increased correlation.









Correlation used from top to bottom: 0.0, 0.2, 0.5, and 0.7.

Higher order polynomial cost estimating relationship with three input variables. $X^5 * X^4 * X^3$, where $E[X]=1$.

Input ranges were +20% / -10%.

Using the same model parameters and changing only the correlation, the estimator can see the impact of correlation on the Monte Carlo output. Markers were used to identify the 50^{th} and 80^{th} percentiles. Note that the X axis is not to scale between model iterations. However, the change in skewness is clearly evident.

50^{th} percentile value consistently decreases while the 80^{th} percentile value consistently increases. Upper and lower bounds of the probability density functions grow as correlation increases.

Results Comparison

The results in general are comparable between Monte Carlo and Method of Moments. The standard deviation and coefficient of variation are generally higher in Method of Moments for an estimate with multiple cost elements using higher order cost estimating relationships. The results may differ with fewer cost elements or lower order cost estimating relationships. The primary cause of difference is that the mean and standard deviation equations in Method of Moments ensure the most pessimistic and most optimistic results are included, whereas with Monte Carlo simulation, the extreme outcomes are left to stochasticity.

Risk Spread Methodologies

Regardless of the risk / uncertainty methodology used, the risk dollars need to be allocated across a program. The risk dollars needed may differ by program phase, program requirements, inherent acquisition milestone risks, or all of the above. Risk allocation is an art in and of itself, with some science involved along with a little bit of luck. Ideally, risk money allocation would be tied to a formal risk assessment. As risks are identified and quantified, risk dollars would be allocated to a program in a timely manner to address the risks identified. However, in practicality, the allocation may be left to estimator judgement.

There are numerous risk allocation methodologies. In this paper, four will be discussed:

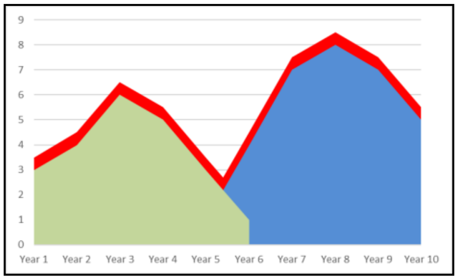
Uniform allocation: constant level of risk dollars applied throughout program

Weighted allocation: variable level of risk dollars applied, based on expenditure profile

Targeted allocation: risk dollars applied at target event based on risk assessment

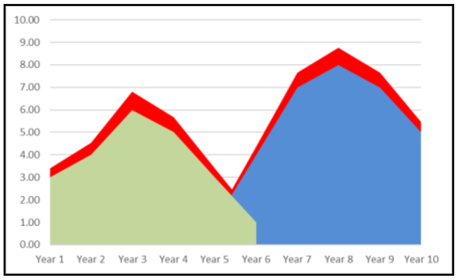
Distribution allocation: front loading or back loading risk dollars using skewed distributions

The weakest of all methodologies is the uniform allocation. This method simply establishes a risk dollar value using the either of the analysis methods described earlier and divides the total risk over the program timeline evenly. The uniform allocation is perhaps the easiest to do mathematically. However, it ignores the nature of risk in a program, particularly the unique risks associated with different phases.



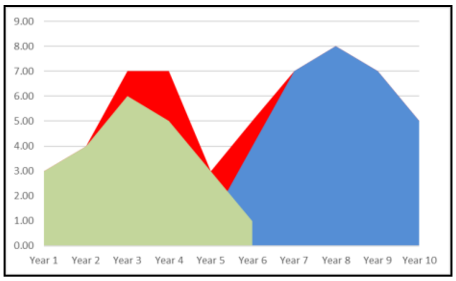
The graph to the left shows typical development (green) and production (blue) acquisition profiles. The risk dollars (red) are evenly spread across the entire program, regardless of phase, milestone, or level of effort.

The weighted allocation method is more targeted than the uniform allocation method. Risk allocation is distributed across the entire program, but the amounts vary based on the amount of expenditures or effort across time. The underlying assumption is that the more a program is executing at any given time, the more likely unforeseen events may surface, causing a need for funds to mitigate the issue.



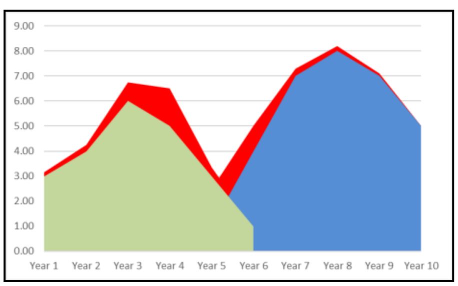
The graph to the left shows typical development (green) and production (blue) acquisition profiles. The risk dollars (red) are spread across the program based on level of effort occurring at any given time.

The targeted allocation employs findings from a formal risk assessment or leverages historical data on risk prone events in a program lifecycle. Risk dollars are applied to key points in the program schedule in an effort to prepare for and mitigate high risk activities such as development testing or initial low rate production.



The graph to the left shows typical development (green) and production (blue) acquisition profiles. The risk dollars (red) are targeted towards high risk activities in the program.

The distribution allocation combines the weighted and event driven methods. In distribution allocation, risk dollars are spread across a phase or program using skewed or normal distribution parameters. Dollars may be back loaded, front loaded, or center loaded in a program, based on perceived or identified risk. This method is easy to automate via spreadsheet tools. It ensures some level of risk dollars are available throughout the program, while key areas are targeted for higher risk allocation.



The graph to the left shows typical development (green) and production (blue) acquisition profiles. The risk dollars (red) are distributed using normal or skewed distribution parameters.

There are numerous ways to automate the processes described in this presentation. Several commercial applications were mentioned. No doubt there are countless spreadsheet-based "home grown" approaches in use throughout the aerospace and defense industry. Regardless of how easy the method is to execute, it is incumbent upon the cost estimator to understand the process behind the method

While there is no one correct way to allocate risk dollars to every program, some methods are better than others. A thorough risk assessment and risk / uncertainty analysis will go far in ensuring the program manager has the right "color" of money, the right amount of money, at the right time.

About the author

Mr. Joe Bauer is a Solutions Consultant with PRICE Systems. He is the primary technical focal point for Air Force customers, providing training, mentoring, and consulting. In addition to the Air Force, Joe supports several defense contractors in the US, as well as key government / defense agencies in Canada. Joe joined PRICE Systems after twenty years of service in the US Air Force. Prior to joining PRICE Systems, Joe was the lead hardware estimator for the F-22 Raptor program office. Joe earned a Master of Science degree in Cost Analysis from the Air Force Institute of Technology in 2009. He earned an MBA from the University of Phoenix in 2005. Joe is also a Certified Cost Estimator / Analyst (CCEA) with the International Cost Estimating and Analysis Association (ICEAA). He can be contacted at Joe.Bauer2@pricesystems.com.

References

1. Introduction to Monte Carlo. Princeton University. Retrieved on March 16, 2018 from http://www.phys.ubbcluj.ro/~zneda/edu/mc/mcshort.pdf.

2. Monte Carlo Method. Wikipedia. Retrieved on March 21, 2018 from https://en.wikipedia.org/wiki/Monte_Carlo_method.

3. Moments (Mathematics). Wikipedia. Retrieved on March 21, 2018 from https://en.wikipedia.org/wiki/Moment_(mathematics).

4. Quick Guide to TruePlanning Uncertainty Analysis. PRICE Systems, LLC. April 1, 2016.

5. Ibid.

6. Ibid.