Presented at the ICEAA 2025 Professional Development & Training Workshop - www.iceaaonline.com/atl2025

MITRE | SOLVING PROBLEMS FOR A SAFER WORLD®

Generating Full Monte Carlo S-Curves Without External Software

AUTHORS:
KEVIN CINCOTTA
TRAVIS GOODWIN
MATT GRIESBACH

March 13, 2025

Approved for Public Release; Distribution Unlimited. Public Release Case Number 25-0876
These technical data were produced for the U. S. government under contract No. FA8702-19-C-0001, and are subject to the Rights in Technical Data-Noncommercial Items Clause DFARS 252.227-7013 (FEB 2014)
© 2025 THE MITRE CORPORATION. ALL RIGHTS RESERVED

MITRE | SOLVING PROBLEMS FOR A SAFER WORLD®

Generating Full Monte Carlo S-Curves Without External Software

Background

This paper presents a cost risk simulation method developed within MITRE's Investment Value Management Framework (IVMF) model. MITRE developed the IVMF model to support sponsors in making informed investment decisions and proactively monitoring the performance of implemented investments throughout the lifecycle. Currently, MITRE is collaborating with a DoD program office to integrate robust investment management principles throughout the entire lifecycle of capability development within the programs.

The development of the IVMF was driven by the recognition that effective investment decisions require a holistic and dynamic approach. The IVMF differs from other modelling capabilities by offering a comprehensive and customizable framework with built-in analytic tools that empower decision-makers to make informed investment choices throughout the investment lifecycle. It also enables tracking of investment performance based on key value drivers specific to each project. Traditional investment decisions often focus on achieving specific milestones and deliverables, rarely revisiting the underlying data of past analyses, which can overlook broader strategic objectives and long-term value. The IVMF seeks to address this gap by:

- **Establishing a common framework:** Providing a consistent and standardized approach to investment management across all programs that incorporate cost, benefit and risk impacts.
- **Fostering collaboration:** Enhancing communication and coordination between stakeholders, including sponsors, program managers, and analysts.
- **Improving decision-making:** Enabling data-driven decisions by providing tools and methodologies for assessing investment costs, non-monetary benefits, and risks.

The IVMF facilitates a comprehensive and iterative approach to investment management by offering the following key functionalities to users:

- **Comparative analysis:** Evaluating alternative investment options based on various criteria, including cost, benefits, risks, and strategic alignment.
- **Iterative refinement:** Supporting continuous improvement of cost estimates, non-monetary benefit assessments, and risk evaluations throughout the program lifecycle.
- **Historical tracking:** Monitoring actual costs, benefits, and risks against initial estimates and forecasts.

By providing a centralized platform for data entry, analysis, and reporting, the IVMF empowers decision-makers with the information needed to make informed and strategic investment choices. The IVMF contains best-of-breed methods for conducting cost assessments, non-monetary benefit assessments, and risk assessments for comparing alternatives. Users enter data, and the model handles the calculations, generating reports that visualize potential outcomes and trade-offs. The IVMF provides a structured model for users to input data, including cost estimates, performance metrics, and risk factors. For example, users can compare different investment options side-by-side, considering both cost and non-cost factors, and see how different risk scenarios might impact the overall value. Critically, the model tracks which inputs have the biggest impact on value, so the user can closely monitor those areas over time and update them as information becomes available. As more data become available and requirements are refined, stakeholders can update the model with actual values and/or refined assumptions. This iterative process allows for continuous validation and refinement of the decision-making process, verifying the rigor of decisions as assumptions evolve into actual results. To ensure efficiency and effectiveness, the model is designed as a self-contained system where users enter or update data and view results in a single Excel file.

Incorporating cost risk and uncertainty analysis (CRUA) directly into the IVMF ensures that decision-makers can assess the financial impact of uncertainty on investment decisions without requiring additional software tools. The flexibility and integrated capabilities of the IVMF model allows for scenario-based analysis (e.g., changing multiple input parameters) and/or sensitivity analysis (e.g., changing one input value) directly in the tool. This enables stakeholders to test different assumptions and refine cost estimates dynamically, as well as see the overall impact to investment value immediately without integrating with other workbooks or analyses.

To facilitate the operational capabilities required to make the IVMF a success, it leverages algorithms that have not yet been exploited in the CRUA community. In this paper, we also describe a new approach to running cost simulations with correlated cost inputs and re-distributing "risk-dollars". We leverage the Normal-To-Anything (NORTA) algorithm[1] to generate correlated random numbers and percentile optimization to redistribute risk-dollars. Our results show these algorithms work at least as well as existing methods and have desirable characteristics. These algorithms are self-contained in the IVMF, which facilitates the execution of CRUA with correlated random numbers without any external software. In particular, the method used to generate correlated random numbers is achieved using only a single custom Excel function, such that correlated random numbers to drive cost simulations can be generated without the use of anything other than one additional Excel Workbook Function and the formula bar.

The following paragraphs are divided into four sections. The first section provides a quick discussion on CRUA application in the IVMF, the overall importance of CRUA and different types of CRUA methods. The second section focuses on a type of CRUA, Monte Carlo simulation, that the IVMF uses. A description of the algorithms leveraged by the IVMF in its implementation is provided. The third section presents numerical results conducted to compare the performance of NORTA and the percentile optimization method. The fourth and final section summarizes the paper and discusses current work to further improve the IVMF to facilitate dynamic percentile optimization based on user prioritization of cost elements and facilitate cost simulation using non-linear copulas to achieve relational correlations.

[1] *Modeling and Generating Random Vectors with Arbitrary Marginal Distributions and Correlation Matrix*. Cairo, Nelson (1997)

Cost Uncertainty in the IVMF

A key component of effective investment management is understanding cost uncertainty and its impact on program outcomes. The IVMF incorporates multiple methods for integrating cost uncertainty into decision-making. The gold standard for cost risk analysis is Monte Carlo simulation, which evaluates the cost drivers of an estimate, and this method of choice is the primary means of representing cost uncertainty within the IVMF. Simulation is a widely accepted practice and can be implemented as either an "inputs"-based simulation "outputs"-based simulation, or combination there-of.[2] Cost analysts can perform this analysis using specific software (e.g., ACEIT), as well as Excel add-ins (e.g., @Risk, Crystal Ball). However, because the IVMF model is self-contained within a single file, it cannot rely on external tools. MITRE developed an embedded CRUA simulation capability using Excel Visual Basic for Applications (VBA), providing a flexible and efficient method for conducting comprehensive cost risk analyses. This cost risk simulation capability within the IVMF model is designed to integrate seamlessly into investment analysis workflows. To enable easy and user-friendly input of data that feeds these automated workflows, VBA scripts facilitate guided user input of required cost data.

Importance of Capturing Cost Uncertainty in Cost Estimates

The only thing that is certain about a cost estimate is that it's certainly wrong. This fact has underpinnings in probability and statistics. Cost can be thought of as a continuous random variable. The probability that a continuous random variable takes on any particular value is zero[3]. Stated differently, cost must be nonnegative, but can take on infinitely many positive values, so the probability that it will exactly equal a given estimate is trivially small. There is also a systemic tendency for cost analysts/estimators and (more so) Subject Matter Experts (SMEs) to understate both the *central tendency* of total cost (Risk[4]) and the *dispersion* around that cost (Uncertainty).

Effective investment decision-making necessitates a comprehensive understanding of potential risks, with cost uncertainty being a paramount factor. Cost uncertainty analysis informs the user of potential variation in budgetary outcomes. Understanding and incorporating cost uncertainty is essential for effective decision-making, as risk always exists that cost estimates driving a program budget may not be completely accurate. Failing to adequately account for cost uncertainty can have significant detrimental impacts on investment outcomes. These detrimental impacts are summarized as follows:

- **Budget Overruns and Program Failure[5]:** Underestimating costs can lead to severe budget overruns, potentially jeopardizing project success. Insufficient funding may necessitate project delays, scope reductions, or even complete abandonment, resulting in substantial financial losses and potentially irrecoverable damage to the mission.

---

[2] *Joint Cost and Schedule Risk and Uncertainty Handbook*. Department of Defense (DoD).

[3] https://sites.nicholas.duke.edu/statsreview/continuous-probability-distributions/

[4] In associating Risk with a shift in central tendency of Cost, we are following the Cost Estimating Body of Knowledge (CEBoK) convention. The GAO Guide and other sources define risk and uncertainty differently—as the probability of occurrence of adverse event.

[5] While COA based cost analysis are not generally used for budgeting, they often are a reference of a starting point for developing a budget after the COA is selected.

- **Erroneous Project Selection:** Making investment decisions without accounting for cost risk uncertainty leaves decision-makers without a clear understanding of the potential variability in outcomes. While point estimates should be unbiased—neither optimistic nor pessimistic, as outlined in the GAO Cost Estimating Guide[6]—ignoring uncertainty prevents decision-makers from assessing their risk tolerance effectively. Without this context, even an accurate point estimate may fail to capture the full range of possible cost outcomes, leading to project selections that do not align with the organization's risk appetite. This can ultimately result in investments that fall short of expectations, misallocating resources, and hindering strategic objectives.
- **Increased Project Risk:** Neglecting cost uncertainty significantly increases the overall risk of project failure. Potential consequences include:
    - **Schedule delays:** Cost overruns often lead to delays in project timelines, impacting project deliverables and potentially disrupting subsequent phases.
    - **Performance issues:** Inadequate funding can compromise the quality of work, impacting project performance and potentially leading to suboptimal outcomes.
    - **Contract disputes:** Budget overruns can lead to disputes with contractors and subcontractors, potentially resulting in costly contract modifications.
    - **Erosion of Stakeholder Confidence:** Frequent budget overruns and project failures can erode the trust and confidence of stakeholders.

Here is a simple example of the presented concepts that also helps demonstrate the distinction between Risk and Uncertainty and provides context to the need of CRUA. Suppose the distribution of a cost is triangular (1,2,4)—meaning a Low (minimum) value of $1, a Most Likely value of $2, and a High (maximum) value of $4. The Point Estimate (PE) of cost is assumed to represent the most likely value[7]. Therefore, in the absence of CRUA, the PE would be $2.00.

Now, CRUA is applied. The mean of a triangular distribution is the mean of its parameters. The mean of these number (to two decimals) is 2.33. The shift in the estimated cost from the PE value of $2.00 to the risk-adjusted mean of $2.33 is referred to as *Risk*. The increase in dispersion (from a fixed value with no dispersion to a number that could be anywhere between $1.00 and $4.00) is referred to as *Uncertainty*. In other words, Uncertainty refers to both the amount of dispersion around the center *and* the shape/distribution of that dispersion. Moreover, Risk Dollars refers to the difference between the risk-adjusted estimate and the PE. This is summarized in Table 1 below.

---

[6] GAO Cost Estimating and Assessment Guide
[7] GAO Cost Estimating and Assessment Guide

Table 1: Risk and Uncertainty for a Hypothetical Cost Element

TYPE OF ESTIMATE	CENTRAL TENDENCY	VARIATION	DISTRIBUTION
PE	$2.00	None	None
Risk and Uncertainty-Adjusted	$2.33	Could take on any value between $1.00 and $4.00	Triangular (1,2,4)
Difference Between Above Rows Referred to as…	Risk	Uncertainty	Uncertainty

Types of Risk and Uncertainty Analyses

CRUA aims to generate an "S-Curve," a graph of the Cumulative Distribution Function (CDF) of total cost, which facilitates the capability to derive a "confidence level" for a given cost magnitude for a specified cost element. This S-Curve, often resembling a standard normal CDF, helps decision-makers estimate the probability of sufficient funding. Several CRUA methods exist to derive this artifact, including input-based (considering risk in cost-driving inputs), output-based (focusing on statistical uncertainty in cost-input relationships), direct S-Curve specification (using methods like Method of Moments or Enhanced Scenario-Based Method (ESBM)[8] to define the S-Curve shape), and risk cube/discrete risks (assessing and monitoring risks as binary events). Range-based analysis, an informal approach, establishes cost boundaries by considering extreme values for each cost element. More information on these methods can be found in the DoD *Joint Cost and Schedule Risk and Uncertainty Handbook*[9], GAO *Cost Estimating and Assessment Guide*[10] and the Cost Estimating Body of Knowledge (CEBoK).[11]

[8] *Enhanced Scenario-Based Method for Cost Risk Analysis: Theory, Application, and Implementation*. Garvey, et. al. (2012)
[9] *Joint Cost and Schedule Risk and Uncertainty Handbook*. Department of Defense (DoD).
[10] U.S. Government Accountability Office (GAO). (2020). *Cost Estimating and Assessment Guide: Best Practices for Developing and Managing Program Costs* (GAO-20-195G)
[11] International Cost Estimating and Analysis Association (ICEAA). (2024). *Cost Estimating Body of Knowledge (CEBoK)*

Implementing Monte-Carlo Simulations

Monte Carlo simulations on the inputs of cost models have become the gold standard for CRUA due to their ability to provide a comprehensive assessment of project's cost risks[12]. By repeatedly sampling from probability distributions assigned to uncertain cost inputs, these simulations generate a large number of potential project outcomes, revealing the likelihood of cost overruns and the potential range of total project costs. The IVMF implements this approach and provides the user with guides, considerations, and benchmarks that assist the user in conducting the analysis and interpreting the results.

A key aspect of accurate Monte Carlo simulations is the proper handling of correlations between cost inputs. In real-world projects, cost uncertainties are often interrelated. Simulations are driven by the ability to generate pseudo-random numbers that mimic independent outcomes. These can be easily generated using spreadsheet software like Excel. However, if these random numbers are not correlated to reflect the dependencies between cost inputs, the simulation may produce overly optimistic results. This is because uncorrelated cost variances tend to cancel each other out, leading to a narrower range of potential outcomes than would naturally be observed. So, without that ability to effectively replicate correlation structures, simulation outcomes may not be completely useful for CRUA[13]. The interested reader can be directed to the corresponding footnote below for further discussion. Of particular importance to this discussion is the DoD Joint Cost Schedule Risk and Uncertainty Handbook (JCSRUH), which emphasizes that considering correlation is crucial for realistic risk assessments.

Roughly speaking, the DoD JCSRUH associates low, moderate, and high-risk efforts with S-Curve CVs of 10%, 30%, and 50%, respectively[14]. In the example shown below (Figure 0-1), we assume that there are 30 cost elements whose costs are normally distributed, with a means of 10 and standard deviations of 3. This implies an entirely reasonable CV of 30% for each element—associated with moderate risk. If correlation is ignored (the elements are treated as independent), the distribution of their sum (total cost) will have a CV of only 5.5%, which the handbook (and any reasonable estimator) would regard as unrealistically low.

---

[12] *Cost Estimating Body of Knowledge* (CEBoK) version 1.2, Risk Module, showing "Input-Based Monte Carlo Simulation" at top of hierarchy/scoring framework.

[13] *Impact of Correlating CER Risk Distributions on a "Realistic" Cost Model*. Smith, Hu (2003).

[14] *Joint Cost and Schedule Risk and Uncertainty Handbook*. Department of Defense (DoD).

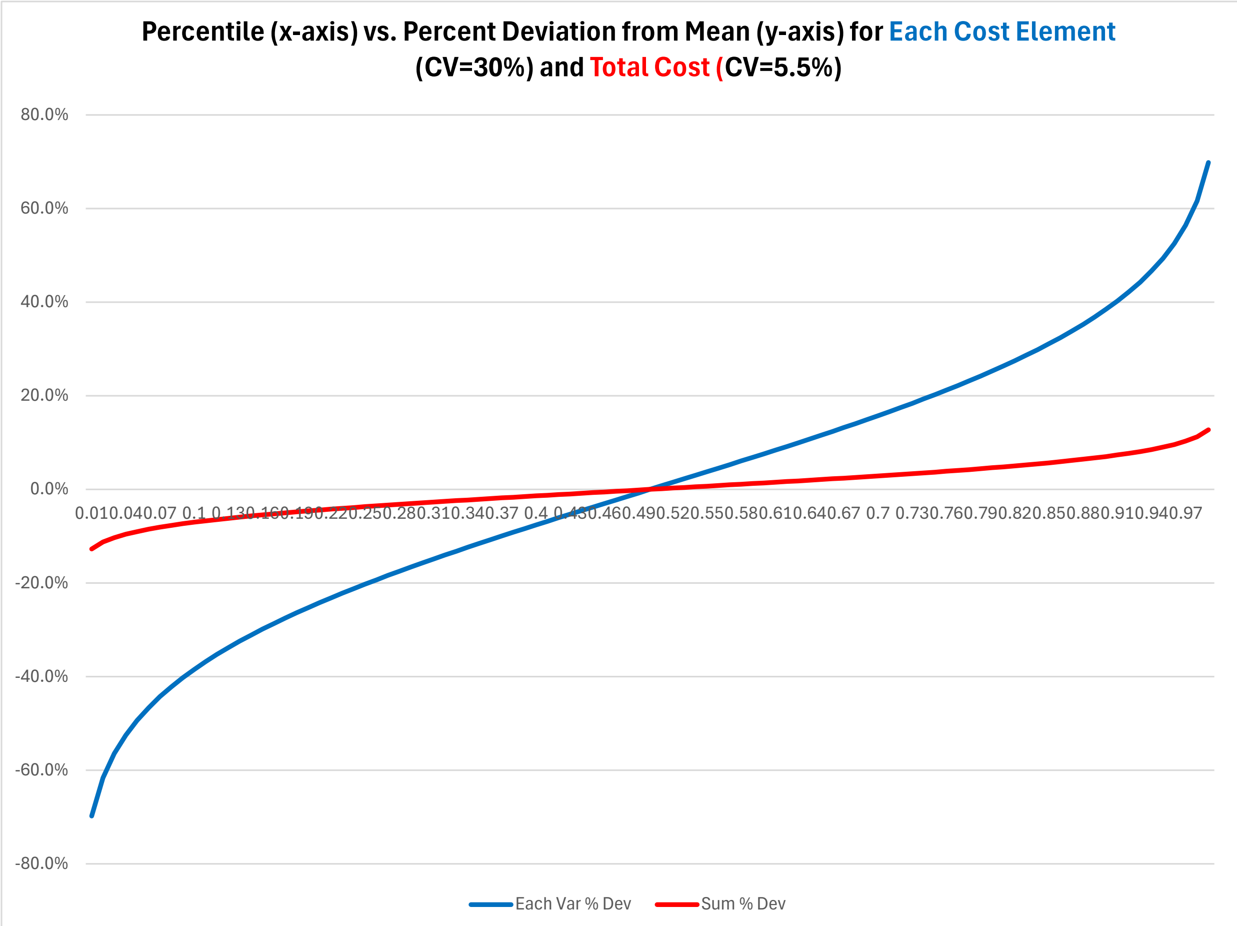


Figure 1 - CV of Summation of 30 Independent Normally Distributed Cost Elements

A CV of only 5.5% is far below the expected 30% for a moderate-risk effort, which is clearly unrealistic. With truly independent variables, as the number of summed variables increases, the combined CV decreases. The total CV of a sum of independent variables can be calculated with the following formula:

$$CV_{total} = \frac{\sqrt{\sum_{i=1}^{n} \sigma_i^2}}{\sum_{i=1}^{n} \mu_i^2}$$

Where:

- CV_{total} is the coefficient of variation for the sum of variables.
- CV_{total} is the standard deviation of each individual variable.
- CV_{total} is the mean of each individual variable.
- n is the number of independent variables being summed.

The variance reduction associated with summing independent (or only lightly correlated) variables is well-known and is associated with the Law of Large Numbers, the Central Limit Theorem, and the Portfolio Effect. We like to call it the Casino Effect, because a casino can experience very high risk/uncertainty on any individual "hand" or roll of the dice/roulette wheel, but there is very little uncertainty in the amount of "house advantage" over the long run. Whatever we call it, this effect is real and must be considered when creating a cost estimate

Arguably the single most important consideration which causes this requirement for estimators to ensure variability is "large enough" for the projection to be realistic is cost correlation. In reality, cost elements are not independent—they are often correlated due to shared cost drivers (e.g., inflation, labor costs, supply chain disruptions). When proper correlation is applied, the total uncertainty is much higher in the simulation output, providing a more realistic estimate.

Generating Correlated Random Variables

The previous discussion should elucidate to the reader the real need for correlation while generating random variables to produce cost distributions that can actually be trusted to make decisions with. A simple way to generate normal variables with a specified correlation is by transforming independent normal variables using an algebraic formula. This method constructs a new variable as a weighted combination of an independent variable and a new random component. This guarantees that the variables maintain a desired correlation while remaining normally distributed.

- **Step 1: Generate Independent Standard Normal Variables:** Let:
  - X_1, X_2, X_3 ,…, X_n be independent standard normal variables, each following N(0,1).

These variables are uncorrelated, meaning they have a correlation of zero with one another.

- **Step 2: Construct a Correlated Variable:** To introduce correlation, **r,** between two variables, we define:

$$X_{new} = rX_1 + \sqrt{1 - r^2} \cdot X_2$$

This transformation ensures that:

- X_{new} remains normally distributed.
- X_1 and X_{new} have a correlation of r.

For example, if **r** = 0. 5:

$$X_{new} = 0.5X_1 + \sqrt{1 - (0.5)^2} \cdot X_2 \Rightarrow 0.5X_1 + \frac{\sqrt{3}}{2}X_2$$

Extending this approach to more variables follows a similar pattern, introducing correlation while preserving normality.

- **Step 3: Scale and Shift to Desired Mean and Standard Deviation:** If the goal is to generate correlated variables with specific means m_i and standard deviations s_i, we apply:

$$Y_i = m_i + s_i \cdot X_i$$

where:

- $Y_i \sim N(m_i\,,\ s_i^2)$
- The correlation structure is preserved.

- **Step 4: Generate Normally Distributed Correlated Samples in Excel:** To implement this in Excel:

  1. In column **A**, generate samples of **X1** using = NORM.INV(RAND(), 0, 1)
  2. In column **B**, generate samples of **X2** using = NORM.INV(RAND (), 0, 1)
  3. In column **C**, compute X_{new} using = r*A1 + SQRT(1-r^2)*B1
  4. In column **D**, compute Y_1 as = m_1 + s_1*A1
  5. In column **E**, compute Y_1 as = m_2 + s_2*C1

This method can be extended to more variables by applying the same transformation iteratively, but the formula's become increasingly more complex. In a three-variable example, the formula for $X_{new,2}$ stays the same, but the formula for $X_{new,3}$ needs to account for correlations between the 3 variables (r_{12} , r_{13}, r_{23}).

$$X_{\text{new},3} = r_{13}X_1 + \frac{r_{23} - r_{12}r_{13}}{\sqrt{1 - r_{12}^2}}X_2 + \sqrt{1 - r_{13}^2 - \left(\frac{r_{23} - r_{12}r_{13}}{\sqrt{1 - r_{12}^2}}\right)^2}\,X_3$$

The increasing complexity of this approach does not make it suitable for use when there are many variables. Fortunately, methods exist which allow us to generate correlated variables at scale in a computationally efficient manner. Up to this point, two existing methods have been considered the current state-of-the-art. One method involves using Spearman Rank Correlation[15]. With this method, uncorrelated random variables are generated from the specified distribution for each cost element. Following this, various algorithms are implemented to match random variables from different iterations across cost elements to attempt and replicate the desired correlation matrix.

A second method involves using *group strength* and is associated with Automated Cost Estimating Integrated Tools (ACEIT) [16]. This method, in its application cited in the footnote, uses

[15] *Population Models and Simulation Methods: The Case of the Spearman Rank Correlation*. Astivia, Zumbo (2017)

[16] *Impact of Correlating CER Risk Distributions on a "Realistic" Cost Model*. Smith, Hu (2003).

a modified Lurie-Goldberg method[17]. The method is presented a vector of “group strengths” associated with how correlated different groups of cost elements are with each other. The Lurie-Goldberg method is modified to use this vector to derive correlated random variables with the relative “group strengths.” Interestingly, the paper introducing the Lurie-Golberg method contains the exact same random number sampling scheme as that presented earlier in NORTA, except that the Lurie-Goldberg method includes a transformation aspect that ensures the provided correlation matrix is perturbed as required to ensure the matrix is positive semi-definite.

Though these methods empirically have been shown to work quite well with consistent correlation matrices, they lack some statistical guarantees that would be desirable from a random number generation method, cannot be implemented without complex VBA code that typically requires the use of Commercial-off-the-Shelf (COTS) software, and (in some cases) cannot be extended beyond linearly independent relationships.

The IVMF leverages NORTA. NORTA solves all three problems. First, its use in generating random variables can provide certain statistical guarantees across any probability distribution (not just normal) in its ability to create pseudo-random number streams that reflect linear correlations relationships while preserving underlying distributions[18]. Second, our outputs only tool (which implements only the Monte Carlo simulation part of the IVMF) can be run without any COTS software and require only a single additional custom excel formula function. Finally, our method can be generalized to include non-linear correlation relationships (as it is a copula-based method) to achieve what has been referred to in the cost community as “relational correlation”[19].

NORTA

The NORTA method leverages Transformation Theory from modern Probability and Statistics to generate correlated random numbers using a copula-based method. Generally speaking, Transformation Theory states that all CDFs have an inverse[20]. Since CDF’s are defined on the [0,1] interval (e.g., a copula), the concept of NORTA is that if correlated random numbers on [0,1] can be generated for n random variables, then those numbers can be transformed back into any desired distribution that preserves both the prescribed linear correlation structure and the desired statistical distributions. The reference to “normal” in NORTA refers to the fact a normal distribution is used to generate the desired correlation structure / copula at the beginning of the algorithm. The choice of a normal distribution is based on the desirable computational attributes of normal random variables.

For a more detailed discussion on NORTA and its characteristics, the reader is referred to footnote 18. The general procedure is as follows:

1. Decompose the desired correlation matrix using the Cholesky Decomposition.

---

[17] *An Approximate Method for Sampling Correlated Random Variables from Partially-Specified Distributions*. Lurie, Goldberg (1998)

[18] *Modeling and Generating Random Vectors with Arbitrary Marginal Distributions and Correlation Matrix*. Cairo, Nelson (1997)

[19] *A Copulas-Based Approach to Modeling Dependence in Decision Trees*. Wang, Dyer (2012).

[20] *Probability*. Karr (1993)

2. Generate i independently and identically distributed (IID) random variables for n standard normal distributions.
3. Multiply the $i \times n$ matrix by the $n \times n$ Cholesky matrix, resulting in a matrix X that contains i replications for n variables.
4. For each of the n random variables with arbitrary marginal distributions, compute $F_{n'}^{-1}(\Phi(X[i', n'])$ for each replication $i' = 1 \dots i$ and distribution $n' = 1 \dots n$.

The primary driving function of this algorithm is the application of Cholesky decomposition. To illustrate how NORTA works in detail, the following section considers a scenario with 3 correlated random variables. With such a small case, the computational details that demonstrate how the algorithm works can be explicitly shown. In the IVMF, unlike the Lurie-Goldberg method, we leverage the Cholesky Full Pivot decomposition algorithm[21], such that any feasible correlation matrix can be adequately represented without adjustment.

Cholesky Decomposition of Covariates

Cholesky decomposition is a process that decomposes a matrix of target correlations (or positive definite matrix) into the product of a lower triangular matrix and its transpose.[22] In a 3-covariate example where the three covariates are correlated as shown in the target correlation matrix $\boldsymbol{\Sigma}$:

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.5 & 0.3 \\ 0.5 & 1 & 0.4 \\ 0.3 & 0.4 & 1 \end{bmatrix}$$

When applying the Cholesky decomposition, we want to find matrix **L** such that:

$$\Sigma = L \cdot L^T$$

Where $\boldsymbol{L}$ is a lower triangular matrix, which means all values above the diagonal are zero. Let $\boldsymbol{L}$ be:

$$L = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix}$$

The **transpose** of $\boldsymbol{L}$, denoted $\boldsymbol{L^T}$, is obtained by swapping rows with columns. Thus, for the given matrix $\boldsymbol{L}$, the transpose $\boldsymbol{L^T}$ will be:

$$L^T = \begin{bmatrix} L_{11} & L_{21} & L_{31} \\ 0 & L_{22} & L_{32} \\ 0 & 0 & L_{33} \end{bmatrix}$$

Thus matrix $\boldsymbol{\Sigma}$ is equal to:

[21] *On the Low Rank Approximation by the Pivoted Cholesky Decomposition.* Harbrecht et. al. (2012).

[22] The **Cholesky decomposition** is a mathematical technique in linear algebra that decomposes a positive-definite matrix into the product of a lower triangular matrix and its transpose. It was introduced by André-Louis Cholesky and published posthumously in 1924. One can equivalently use an upper triangular matrix, which is done in our slides.

$$\Sigma = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \cdot \begin{bmatrix} L_{11} & L_{21} & L_{31} \\ 0 & L_{22} & L_{32} \\ 0 & 0 & L_{33} \end{bmatrix}$$

Performing the matrix multiplication step-by-step yields:

$$\Sigma = \begin{bmatrix} {L_{11}}^2 & L_{11}L_{21} & L_{11}L_{31} \\ L_{11}L_{21} & {L_{21}}^2 + {L_{22}}^2 & L_{21}L_{31} + L_{22}L_{32} \\ L_{11}L_{31} & L_{21}L_{31} + L_{22}L_{32} & {L_{31}}^2 + {L_{32}}^2 + {L_{33}}^2 \end{bmatrix}$$

Now, we set this product equal to the target correlation matrix $\boldsymbol{\Sigma}$:

$$\begin{bmatrix} {L_{11}}^2 & L_{11}L_{21} & L_{11}L_{31} \\ L_{11}L_{21} & {L_{21}}^2 + {L_{22}}^2 & L_{21}L_{21} + L_{22}L_{32} \\ L_{11}L_{31} & L_{21}L_{31} + L_{22}L_{32} & {L_{31}}^2 + {L_{32}}^2 + {L_{33}}^2 \end{bmatrix} = \begin{bmatrix} 1 & 0.5 & 0.3 \\ 0.5 & 1 & 0.4 \\ 0.3 & 0.4 & 1 \end{bmatrix}$$

Solving for the individual elements of $\boldsymbol{L}$ yields a final matrix for $\boldsymbol{L}$:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 0.5 & \frac{\sqrt{3}}{2} & 0 \\ 0.3 & \frac{\sqrt{3}}{6} & \sqrt{\frac{62}{75}} \end{bmatrix}$$

For a detailed derivation of each element, refer to the Appendix.

The elements of $\boldsymbol{L}$ can be understood as follows:

- $\boldsymbol{L_{11}} = 1$: This represents perfect correlation, or, the relative uncertainty of the random numbers generated for this variable, what we will naively refer to here as the "unit variance" of this variable, conditioned on previously decomposed variables. Since this is the first decomposed column, there is no "information" we are conditioning on, and so this particular variable is considered independently from all other variables in our decomposition and has no change in its correlation structure from the original matrix. Therefore, the IID random numbers and posterior correlated random numbers will be the exact same for this particular variable, and they will have "perfect correlation". In subsequent variables, this "unit variance" number will not be one, because it will be conditioned on knowledge about this variable based on the proposed correlation structure (see below). Therefore, the posterior correlated random numbers will not be the same as the original IID numbers, and the correlation between them will not be 1.
- $\boldsymbol{L_{21}} = 0.5$*:* This is the strength of the linear relationship between the first and second variables conditioned on the fact that the first variable will be independent of all other variables (hence, this value is unperturbed from the original matrix and is equal to the actual correlation value). The value 0.5 indicates a moderate positive correlation between them.

- $\boldsymbol{L_{22}} = \frac{\sqrt{3}}{2}$: This number might be naively considered the "unit variance" of the second set of random numbers *conditioned on knowledge of the first set of random numbers.* Since the first and second variables are correlated, given one knows what the first set of random numbers looks like, it becomes possible to "partially guess" within a certain range of error what the second set of random numbers will look like (think linear regression). The error associated with the fact that these two variables are not perfectly correlated becomes something like the "unit variance" of the second set of random numbers.
- $\boldsymbol{L_{31}} = 0.3$ This represents the linear relationship between the first and third variables conditioned on the fact that the first variable will be independent of all other variables (hence, this value is unperturbed from the original matrix and is equal to the actual correlation value). The value 0.3 indicates a smaller positive correlation between the first and third variables.
- $\boldsymbol{L_{32}} = \frac{\sqrt{3}}{6}$: This is the strength of the linear relationship between these variables conditioned on knowledge of the first variable, given the correlation between the first variable and the second variable as well as the correlation between the first variable and the third variable, similar to L_{22}.

  $\boldsymbol{L_{33}} = \sqrt{\frac{62}{75}}$: This represents the "unit variance" of the third variable, conditioned on knowledge of both the first and second variable, subject to the correlation between the third variable and both the first and second variables. Generally speaking (but with fairly regular exceptions, this being one of them), these diagonal decrease in magnitude and go towards zero. This represents the fact that as knowledge about more variables is gathered in a correlated system, it becomes easier to predict what the next set of variables will look like.

This matrix $\boldsymbol{L}$ is a lower triangular matrix that transforms independent standard normal random variables into correlated random variables. Each element in matrix $\boldsymbol{L}$ represents (in very loose terms) the conditional "unit variance" of a variable and given the conditional correlation between any two variables given the relative magnitude of their "unit variance".

The Cholesky decomposition allows us to model the covariance structure of a set of variables and generate correlated random variables for simulations or risk analysis models. The actual make-up of the Cholesky depends on how the variables are ordered in correlation matrix; so, in the above example, if X_3 were in the first position of the correlation matrix, one would get a different L. However, the overall results for the random number generation would be approximately the same. In the Full Pivot algorithm used in the IVMF, a user will get the exact same results regardless of the ordering because this algorithm orders processing of the Cholesky based on eigenvalue magnitude.

Generating Random Variables

The next step is to generate independent random variables, which follow a standard normal distribution (mean of 0 and standard deviation of 1). These are the base random numbers that will later be transformed into correlated random variables.

Denote these independent standard normal random variables as $\mathbf{Z_1}, \mathbf{Z_2}, \mathbf{Z_3}$. In Excel, use the **NORM.INV(RAND(),0,1)** or **NORM.S.INV(RAND())** function to generate standard normal random numbers. These values are independent of each other. For each iteration, generate three independent random variables $\mathbf{Z_1, Z_2, Z_3}$ that follow a standard normal distribution. These will be the base for generating correlated variables using the Cholesky Matrix.

Given independent standard normal random variables $(\mathbf{Z_1, Z_2, Z_3})$, apply the Cholesky transformation using the lower triangular matrix **L** obtained from decomposing the covariance matrix $\mathbf{\Sigma}$. This transforms the independent random variables into correlated random variables.

For each pair of independent random variables $\mathbf{Z_1, Z_2, Z_3}$, apply the transformation to get the correlated random variables $\mathbf{X_1, X_2, X_3}$.

For $\boldsymbol{X_1}$:

$$X_1 = L_{11} \cdot Z_1 + L_{12} \cdot Z_2\, L_{13} \cdot Z_2 = 1 \cdot Z_1 + 0 \cdot Z_2 + 0 \cdot Z_3 = Z_1$$

$$X_1 = Z_1$$

For $\boldsymbol{X_2}$:

$$X_2 = L_{21} \cdot Z_1 + L_{22} \cdot Z_2 + L_{23} \cdot Z_3 = 0.5 \cdot Z_1 + \frac{\sqrt{3}}{2} \cdot Z_2 + 0 \cdot Z_3$$

$$X_2 = 0.5 \cdot Z_1 + \frac{\sqrt{3}}{2} \cdot Z_2$$

For $\boldsymbol{X_3}$:

$$X_3 = L_{31} \cdot Z_1 + L_{32} \cdot Z_2 + L_{33} \cdot Z_3$$

$$X_3 = 0.3 \cdot Z_1 + \frac{\sqrt{3}}{6} \cdot Z_2 + \sqrt{\frac{62}{75}} \cdot Z_3$$

Now, generate **N** sets of **independent** random variables $\mathbf{Z_1, Z_2, Z_3}$, which are standard normal variables (i.e., they follow a normal distribution with mean 0 and standard deviation 1). To clarify, while the random numbers generated in Excel will be between 0 and 1, this algorithm uses the **inverse normal distribution** function (NORM.INV()) to transform these values into standard normal random variables.

1. **Column A: Generate N independent random variables for $\mathbf{Z_1}$:** Use the formula =NORM.INV(RAND(),0,1) for each row.
2. **Column B: Generate N independent random variables for $\mathbf{Z_2}$:** Use the formula =NORM.INV(RAND(),0,1) for each row.
3. **Column C: Generate N independent random variables for $\mathbf{Z_3}$**: Use the formula =NORM.INV(RAND(),0,1) for each row.
4. **Column D: Calculate $\boldsymbol{X_1}$:** Use the formula = A1. Since $\boldsymbol{X_1} = \mathbf{Z_1}$, you directly reference the value of $\mathbf{Z_1}$ from Column 1.
5. **Column E: Calculate $\boldsymbol{X_2}$:** Use the formula = 0.5*A1 + SQRT(3)/2 *B1. This formula applies the transformation to create the correlated random variable $\boldsymbol{X_2}$ which is a linear combination of $\mathbf{Z_1}$ and $\mathbf{Z_2}$.

6. **Column F: Calculate X_3:** Use the formula = 0.3*A1 + SQRT(3)/6 *B1 + SQRT(62/75) *C1. This formula transforms $\mathbf{Z_1, Z_2, Z_3}$ into X_3, incorporating the correlations between these variables.

The resultant N iterations of the random variables $\boldsymbol{X_1}, \mathbf{X_2}, \mathbf{X_3}$ are now effectively correlated, which were generated from the independent standard normal random variables $\mathbf{Z_1, Z_2, Z_3}$ using the Cholesky decomposition matrix.

This transformation process allows a user to generate correlated random variables for risk analysis or Monte Carlo simulations based on an assumed correlation structure. To summarize:

- After running N iterations, there will be three arrays $\boldsymbol{X_1}$ $\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ of length N each, representing the correlated random variables.
- These variables will have a correlation close to the value specified by the covariance matrix from which the Cholesky decomposition was performed

By repeating this process, you will be able to generate **N** iterations of correlated random numbers that you can use for Monte Carlo simulations.

Percentile Optimization

In addition, the IVMF leverages a new method for re-distribution of "risk-dollars". This new method uses optimization to determine what percentile child-level WBS elements should be budgeted to meet user-specified risk percentiles for total cost. Previous methods for redistribution of risk-dollars have largely been based on a combination of the magnitude of each cost element and its associated covariance (e.g., Book's method[23]).

The IVMF contains an alternative mechanism to perform this task. The user is allowed to specify either a total cost percentile to budget to (e.g., derive the required total cost to meet the 80th percentile of total cost), or to specify a base-level cost element risk percentile (e.g., set all non-rollup cost elements to the 80th percentile). In the latter case, the IVMF will automatically roll up and back-compute the resulting percentile for each roll up item. In the former case, when the user specifies a total budget percentile, an optimization algorithm executes to compute the percentile at which each base level cost element will be budgeted to in order to get the total cost of the project to within a user specified error tolerance of the desired total budget percentile.

In the numerical results section, we compare the output between Book's method and our percentile optimization method and provide a discussion in the final section. Here, we explain the rationale behind the proposed method. "Regularization"[24] is a common computational technique, where-in, the "volatility" of a model output is controlled, or averaged, to reduce "over-fitting" to an observed set of data based on assumptions that may be incorrect. Current methods for re-distributing funds require inherent assumptions about what actual cost element will be more important than others regarding the desired outcome expected from the budget provided. In the percentile optimization method, we assume that no cost element is any more important than the other in the context of the assumed goal achieved by the provided budget. To achieve this, the optimization algorithm in the IVMF budgets each child-level cost element at the same percentile to derive a solution that meets user specified total budget risk. This essentially

[23] *Allocating Risk Dollars Back to WBS Elements*. Book (2007)
[24] *Applied Predictive Modelling.* Kuhn, Johnson (2013)

“regularizes” the distribution of risk-dollars and yields insights to users about alternative implementation plans when assumptions are removed that other methods are based on. Additionally, this method can be amended in the future to allow a user to weight cost elements differently based on perceived importance. In doing so, the user can explore the implications of different assumptions when accounting for risk in a budget.

Numerical Experiments

To test these methods against other known methods, we present the following results from numerical experiments. These experiments were run to evaluate the ability of NORTA and the proposed percentile optimization method against other known methods. Specifically, for NORTA, we compare the speed and accuracy of running NORTA to generate random variables against the Spearman-Rank algorithm in Crystal Ball. For the proposed percentile optimization method, we compare our results with those derived from applying Book's positive semi-covariance method. In each of the following two sections, we describe the experiments that were conducted and state the differences observed between the methods. A discussion of the results and conclusions are included in the final section.

Comparing NORTA with Spearman-Rank Correlation Algorithms

In these results, we compare the performance between two different methods for generating correlated random variables. The two methods tested are the Rank-Ordering algorithm from Crystal Ball and the NORTA method implemented in the IVMF. Performance is measured through two metrics: the time it takes to execute (run-time), and the error associated with the resultant correlation matrix. "Error" is measured through the Frobenius[25] norm, a method for measuring "distance" between matrices, and is a version of the 2-norm (the 2-norm is a p-norm based common loss function used in things like linear regression). The value in the error column should be interpreted as the "distance" between the desired correlation matrix and the resultant correlation matrix. Because "distance" for matrices and vectors is unitless, we must provide a baseline against which to compare these numbers; the "baseline" score is the average distance between the goal correlation matrix and both a matrix of all 1's and a matrix of all 0's. In each experiment, five-thousand (5000) replications are performed to gain a "long-run" estimate with minimal error regarding the performance of each.

Test	Variables	Rank-Order Time (s)	Rank-Order Error	NORTA Time (s)	NORTA Error	Linearly Dependent?
1	2	2.51	0.043	0.19	0.008	N
2	10	4.48	0.131	0.95	0.134	N
3	10	4.51	0.147	0.92	0.160	N
4	10	4.67	0.304	0.93	0.844	Y
5	50	5.11	0.822	5.28	0.734	N
6	50	4.71	13.549	5.27	17.765	Y
7	50	4.52	8.575	4.98	23.461	Y
8	18	3.67	0.379	1.82	0.264	Y

Figure 2 – Experimental Design End Results for Correlated Number Generation

NORTA generally performs better than the rank-order method when the matrix is linearly independent and is generally faster. The Rank Order method performs better when the matrix is linearly dependent. The methods are comparable. However, only the final test contained real-

[25] Trefethen, L. and Bau, D. *Numerical Linear Algebra*. p, 22. 1997, Society for Industrial and Applied Mathematics.

world data. In that test, NORTA outperformed the Spearman-Rank correlation algorithm in Crystal Ball, despite the fact that the matrix was linearly dependent. Of particular note is that this matrix was labelled as "inconsistent" according to the Spearman-Rank algorithm and was required to be transformed prior to random number generation (as opposed to NORTA, which was able to faithfully generate random numbers to the original matrix).

Comparing Percentile Optimization with Book's Method

Here, we compare the results of distributing risk-dollars using the proposed percentile optimization method and Steve Book's method. Whereas Book's method prioritizes WBS elements that have a larger covariance, the percentile optimization method is purely based on the element's distribution. With this approach, the percentiles that result from the re-distribution are more "regularized," in the sense that they do not favor any one element more than another. There are advantages and disadvantages to both methods. Book's method offers insight into how to redistribute money if one assumes that cost elements with a larger magnitude of cost will have more impact on the assumed project outcome from the prescribed budget. The optimization method offers insight into how the best redistribution might look if one doesn't make any assumption about which cost elements would have more impact on the assumed outcome.

Cost Element	Point Estimate	Book's Budget	Percentile	Expected Delta	Optimization Budget	Percentile	Expected Delta
Investment	$2,337,843	$2,600,244	80.0%	-$186,651	$2,600,244	80.0%	-$186,651
Program/Project Management	$30,000	$31,669	55.9%	-$5,568	$35,141	70.1%	-$3,920
Planning Phase Program/Project Management	$15,000	$15,835	55.9%	-$2,781	$17,561	70.1%	-$1,970
Acquisition Phase Program/Project Management	$15,000	$15,835	55.8%	-$2,777	$17,574	70.1%	-$1,957
Systems Engineering (or Systems Analysis)	$134,043	$151,208	63.5%	-$42,507	$164,896	71.4%	-$38,846
Planning Phase Systems Engineering	$104,703	$119,223	63.1%	-$39,976	$128,954	70.1%	-$38,447
Acquisition Phase Systems Engineering	$29,340	$31,985	59.2%	-$10,285	$35,730	70.1%	-$9,637
Business Process Re-engineering (BPR) / Change Management	$599,100	$666,071	69.8%	-$83,129	$677,371	72.7%	-$80,113
Planning Phase BPR/Change Management	$104,355	$111,446	59.9%	-$19,022	$119,042	70.1%	-$16,732
Acquisition Phase BPR/Change Management	$494,745	$554,625	69.2%	-$80,858	$558,271	70.1%	-$79,578
System Development	$300,000	$317,521	64.7%	-$33,163	$339,148	79.9%	-$28,703
Software Development	$30,000	$31,214	60.3%	-$4,251	$32,723	70.1%	-$3,902
Data Development & Transition Planning	$30,000	$31,229	60.2%	-$4,218	$32,757	70.1%	-$3,854
Data Base Standards/Dictionary	$30,000	$31,896	59.8%	-$6,362	$34,185	70.1%	-$5,868
Training Development	$30,000	$31,901	62.2%	-$6,667	$33,941	70.1%	-$6,114
Test and Evaluation	$30,000	$32,631	59.7%	-$8,452	$35,945	70.1%	-$7,548
Logistics Support Development	$30,000	$31,861	59.2%	-$6,271	$34,168	70.1%	-$5,841
Development/Prototype Facilities	$30,000	$31,184	60.2%	-$3,977	$32,546	70.1%	-$3,724
Environmental Studies	$30,000	$31,863	61.8%	-$6,245	$33,847	70.1%	-$5,726
Other Development	$30,000	$31,835	58.4%	-$6,065	$34,534	70.1%	-$5,236
System Procurement	$1,274,700	$1,433,774	**77.3%**	-$136,207	$1,383,675	**70.1%**	-$146,926

Figure 3 – Comparing Risk Dollar Allocation Methods

In this example, our method gives more risk dollars to most elements, budgeting the lowest-level elements at the 70.1st percentile, compared to roughly the 60th percentile for lowest-level elements in Book's method. But Book's method budgets the largest cost element, System Procurement, at the 77.3rd percentile—a big increase over the 70.1st percentile derived from the IVMF. Tables like this help the analyst to see "where the risk dollars are going". In certain contexts, it may make sense to relatively favor the biggest ticket items (Book's method) or to spread the risk dollars relatively more proportionately (our method).

Of course, the optimal way to allocate risk dollars is to not allocate them all and instead hold them in reserve. That way, they can be used whenever and wherever they are needed.

Nevertheless, one best practice in the GAO Guide and elsewhere is to allocate the dollars; this provides insight into two methods for doing so.

While this optimization technique currently does not allow users to weight different cost elements as more important or not (and therefore, assumes equal weight to each cost element, and so assigns the same percentile to each cost element), this is relatively easy to incorporate and planned for in a future version. The IVMF also has the capability to compute the required total budget percentile for a desired child-level element percentile (i.e., use our logic in reverse), and a future version will allow users to run this functionality with varying percentiles across cost elements.

Presented at the ICEAA 2025 Professional Development & Training Workshop - www.iceaaonline.com/atl2025

Discussion and Summary

Failure to conduct CRUA can lead to biased decision-making that inadvertently favors high-risk alternatives. The methodology presented in this paper provides a systematic and mathematically rigorous approach tailored for the cost analyst community. While built upon well-established CRUA and correlation techniques, our approach represents a novel practical implementation that enhances accessibility and analytical consistency.

A key advancement of our methodology is the application of the Cholesky/NORTA approach for capturing correlation. While non-linear copulas are theoretically the most accurate method, to our knowledge, they have not been practically implemented in cost analysis due to their computational complexity and integration challenges. Our method provides a practical alternative to existing capabilities with increased mathematical rigor and improved performance, particularly in the tail regions of the distribution. Compared to Crystal Ball, our approach potentially offers superior handling of extreme cost scenarios, thereby enhancing risk assessments. Additionally, the methodology streamlines the cost risk assessment process and does not rely on third party software or tools. Our method is fully operational within Excel, increasing usability.

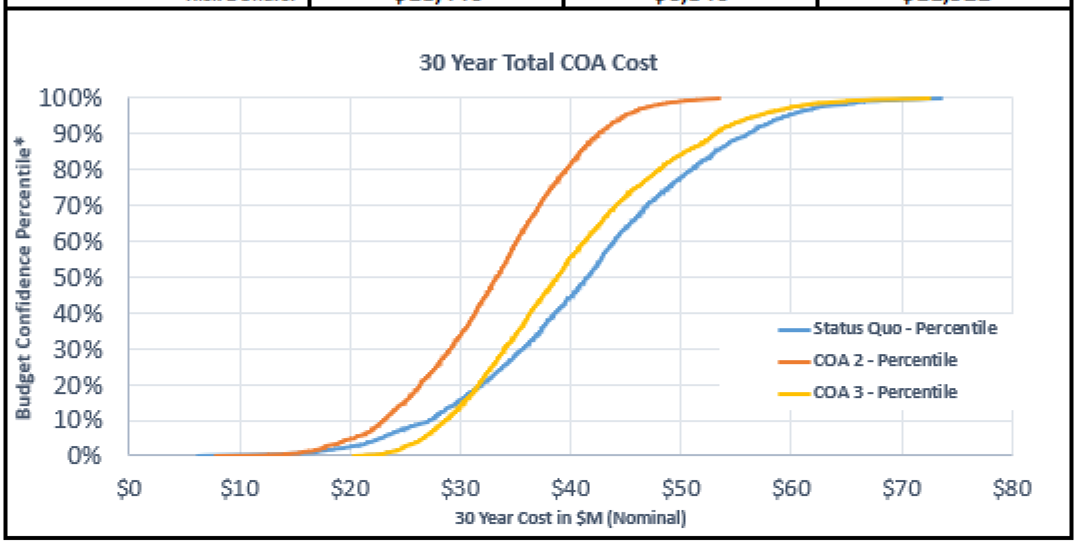
Our method has been implemented in two distinct ways, each designed to accommodate varying analyst needs and levels of complexity. These implementations provide flexibility based on the specific requirements of a given cost estimate.

The first implementation is the Output-Only version, a streamlined and low-code approach optimized for ease of use and rapid analysis. This version allows users to customize the WBS within the file and generate uncertainty cost estimates with minimal setup effort. Users can specify distribution parameters and correlations between cost elements, and the model instantaneously produces S-curves and key statistical measures. This implementation is particularly effective for analysts who require a quick yet robust cost risk assessment without extensive computational overhead. While ESBM can easily and quickly be applied to produce S-Curves on total costs, our Output-Only tool allows users to define distributions for individual cost elements and establish correlation pairs among them, enabling a more refined representation of cost risk uncertainty.

The second implementation integrates our CRUA methodology into the IVMF model, allowing for a more comprehensive uncertainty analysis within a broader comparative analysis. The IVMF automates many commonly performed comparative analysis functions, reducing manual effort and improving analytical consistency. The IVMF implementation offers flexibility and many different modular capabilities, supporting comparative analysis across multiple courses of action (COAs) while incorporating both cost and non-monetary considerations. Analysts can use this approach to conduct full-scale Analysis of Alternatives (AOA) within the file or only perform cost risk assessments. The automation embedded within the IVMF further enhances its usability by streamlining model setup, calculations, and reporting processes. This reduces the analyst's workload while ensuring consistency in the application of CRUA principles.

In this version of the IVMF, users construct their cost models within the file, specifying distribution parameters and correlations for both cost inputs and cost elements. This enables uncertainty analysis at both the input and output levels, making it suitable for detailed cost assessments. Furthermore, the auto-generated reports provide clear visual representations of cost risk through S-curve outputs and statistical comparisons across multiple risk-adjusted cost estimates. These features allow analysts to evaluate different scenarios efficiently and present findings in a structured and interpretable manner. By embedding CRUA within the IVMF, we seamlessly integrate rigorous cost risk assessment into the decision-making process, reinforcing analytical rigor while simplifying execution.

Comparison and Precision Data			
Comparison Percentile at the Cost Element Level or Total Cost Level			Total
Comparison Percentile			80%
Step Size			0.001
Cost Element to Plot:	**Total Cost - $K Nominal**		
Cost Element Name	**Status Quo**	**COA 2**	**COA 3**
Point Estimate (PE):	**$32,500**	**$32,771**	**$36,162**
Percentile of PE:	**21.1%**	**47.1%**	**39.0%**
Minimum:	$3,241	$5,886	$19,596
Median:	$41,525	$33,314	$38,825
Mean:	$41,342	$33,004	$39,938
Percentile of Mean:	49.6%	48.6%	55.4%
Maximum:	$79,210	$55,064	$74,235
Standard Deviation:	$11,156	$7,595	$9,357
CV:	0.27	0.23	0.23
Risk-Adjusted Cost:	**$50,940**	**$39,611**	**$48,084**
Risk Adjustment Percentile:	**80.1%**	**80.1%**	**80.1%**
Percent increase to PE:	56.7%	20.9%	33.0%
Risk Dollars:	**$18,440**	**$6,840**	**$11,922**



*Note - Official Budget numbers require consideration of funding source and may need to be converted to Then Year dollars.

Figure 4 – IVMF S-Curve Comparison Report

Further development will focus on several key areas to enhance the model's capabilities and robustness. Rigorous numerical experiments will be conducted to validate the method across a broader range of scenarios and datasets. This involves testing the model with diverse input parameters, varying complexity levels, and comparing results against established benchmarks or real-world data where available. The goal is to thoroughly assess the model's accuracy, stability, and sensitivity to different factors, identifying any potential limitations and ensuring reliable performance across a wide spectrum of applications.

Incorporating non-linear copula-based correlation modeling is planned to improve the accuracy of dependency representation. While the current Cholesky/NORTA method provides a valuable step in capturing correlation, non-linear copulas offer a more sophisticated approach, particularly for modeling complex dependencies and tail behaviors. For some time, the cost community has discussed that applying non-linear copulas is the superior method for CRUA, but no practical implementation has been developed. Exploring and implementing these advanced techniques will allow the model to more accurately reflect the relationships between variables, leading to more realistic and robust risk assessments.

Enhanced risk dollar allocation capabilities will be explored to provide more detailed insights into cost drivers and schedule. This will involve developing methodologies for attributing risk to specific cost and schedule drivers, visualizing these contributions, and potentially incorporating optimization algorithms to support risk-informed decision-making.

The creation of a standalone IVMF Excel product could broaden accessibility. Currently, the IVMF model contains much more functionality and capability than is required for conducting CRUA. Developing a standalone Excel version would significantly lower the complexity of the file for users that only are interested in the cost analysis capabilities.

Appendix - Solving for Matrix Elements

The following section presents a detailed mathematical derivation to determine the elements of matrix L. Building upon the matrix algebra conducted in the previous section, Cholesky Decomposition of Covariates, we arrived at the following equation:

$$\begin{bmatrix} {L_{11}}^2 & L_{11}L_{21} & L_{11}L_{31} \\ L_{11}L_{21} & {L_{21}}^2 + {L_{22}}^2 & L_{21}L_{21} + L_{22}L_{32} \\ L_{11}L_{31} & L_{21}L_{31} + L_{22}L_{32} & {L_{31}}^2 + {L_{32}}^2 + {L_{33}}^2 \end{bmatrix} = \begin{bmatrix} 1 & 0.5 & 0.3 \\ 0.5 & 1 & 0.4 \\ 0.3 & 0.4 & 1 \end{bmatrix}$$

Solve for the individual elements of $\boldsymbol{L}$:

1. First Equation: $\boldsymbol{L_{11}}$

$${L_{11}}^2 = 1 \Rightarrow L_{11} = 1$$

2. Second Equation: $\boldsymbol{L_{21}}$

$$L_{11}L_{21} = 0.5 \Rightarrow 1 \cdot L_{21} = 0.5 \Rightarrow L_{21} = 0.5$$

3. Third Equation: $\boldsymbol{L_{31}}$

$$L_{11}L_{31} = 0.3 \Rightarrow 1 \cdot L_{31} = 0.3 \Rightarrow L_{31} = 0.3$$

4. Fourth Equation: $\boldsymbol{L_{22}}$

$${L_{21}}^2 + {L_{22}}^2 = 1$$

Substituting $\boldsymbol{L_{21}} = 0.5$

$$0.5^2 + {L_{22}}^2 = 1 \Rightarrow 0.25 + {L_{22}}^2 = 1 \Rightarrow {L_{22}}^2 = 0.75$$

$$L_{22} = \sqrt{0.75} = \frac{\sqrt{3}}{2}$$

5. Fifth Equation: $\boldsymbol{L_{32}}$

$$L_{21}L_{31} + L_{22}L_{32} = 0.4$$

Substituting $\boldsymbol{L_{21}} = 0.5$, $\boldsymbol{L_{22}} = \frac{\sqrt{3}}{2}$ and $\boldsymbol{L_{31}} = 0.3$

$$0.5 \cdot 0.3 + \frac{\sqrt{3}}{2}L_{32} = 0.4$$

$$\frac{\sqrt{3}}{2}L_{32} = 0.25 \Rightarrow \frac{\sqrt{3}}{2}L_{32} = \frac{0.25 \times 2}{\sqrt{3}} \Rightarrow L_{32} = \frac{0.5}{\sqrt{3}}$$

$$L_{32} = \frac{\sqrt{3}}{6}$$

6. Sixth Equation: $\boldsymbol{L_{33}}$

$$L_{31}{}^2 + L_{32}{}^2 + L_{33}{}^2 = 1 \Rightarrow (0.3)^2 + \left(\frac{\sqrt{3}}{6}\right)^2 + L_{33}{}^2 = 1$$

$$0.09 + \frac{3}{36} + L_{33}{}^2 = 1 \Rightarrow 0.09 + \frac{1}{12} + L_{33}{}^2 = 1$$

$$L_{33}{}^2 = 1 - \left(\frac{9}{100} + \frac{1}{12}\right) \Rightarrow L_{33}{}^2 = 1 - \frac{13}{75}$$

$$L_{33}{}^2 = \frac{62}{75} \Rightarrow L_{33} = \sqrt{\frac{62}{75}}$$

$$L_{33} = \sqrt{\frac{62}{75}}$$

These numbers yield our final matrix for $\boldsymbol{L}$:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 0.5 & \frac{\sqrt{3}}{2} & 0 \\ 0.3 & \frac{\sqrt{3}}{6} & \sqrt{\frac{62}{75}} \end{bmatrix}$$

Acronyms and Abbreviations

ACEIT	Automated Cost Estimating Integrated Tools
AoA	Analysis of Alternatives
CDF	Cumulative Distribution Function
CEBoK	Cost Estimating Body of Knowledge
CRUA	Cost Risk and Uncertainty Analysis
CV	Coefficient of Variation
COA	Course of Action
COTS	Commercial-off-the-Shelf
ESBM	Enhanced Scenario-Based Method
GAO	Government Accountability Office
ICEAA	International Cost Estimating and Analysis Association
IID	Independently and Identically Distributed
IVMF	Investment Value Management Framework
JCSRUH	Joint Cost and Schedule Risk and Uncertainty Handbook
NORTA	Normal to Anything
PE	Point Estimate
SME	Subject Matter Expert
VBA	Visual Basic for Applications
WBS	Work Breakdown Structure