BASIC SIMULATION CONCEPTS

INTRODUCTION

Simulation is a technique that involves modeling a situation and performing experiments on that model. A model is a program that imitates a physical or business process by use of a variable that can be manipulated. The model is used in simulation by causing it to respond mathematically to changing conditions as though it were the process or object itself. In simulation, the facility or process to be examined is called the system of interest.

Consider, for example, taxing a retail transaction as a system of interest. We can model this as a simple build-up equation:

\[ \text{Sales Tax} = \text{Price} \times \text{Sales Tax Rate} \]

In order to study the system of interest scientifically, we often have to make assumptions on how it works. The set of assumptions constitutes a model that is used to gain understanding of how the system behaves. If the relationship (that composes the model) is simple enough, it may be possible to use math methods to obtain exact information. This is called an analytic method. Continuing the retail tax example, if the tax rate is known with certainty (say 7% tax rate) and an item’s price is known with certainty (say $100), the model is solved analytically:

\[ \text{Sales Tax} = \text{Price} \times \text{Sales Tax Rate} \]
\[ \text{Sales Tax} = 100 \times 0.07 \]
\[ \text{Sales Tax} = 70 \]

Most real world systems are too complex to be evaluated analytically and these systems are studied by simulation. The use of models allows one to test the outcomes of a proposed group of inputs and processes, prior to their implementation in real life.

Under these circumstances, the model will typically take the form of a computer program that contains the entities or components that represent the elements of the real system of interest. When the computer program is executed, the behavior of the system can be deduced by noting what happens at various points in the simulation. Thus, the computer emulates the behavior of the system and is...
programmed to gather statistics about the behavior of the model. It is through the numeric technique of variable manipulation of the model that the behavior of the system is studied.

Model Building Process
The following basic steps are used in the simulation process.

(1) **Definition of the problem.** For any study, a clear problem statement and the study objectives are important. This step of the procedure should include determining the boundaries, restrictions, and assumptions associated with the problem.

(2) **Model formulation.** The model structure must be established to include only those aspects necessary and relevant to the study objectives. The inclusion of irrelevant information in the model increases the complexity and cost of the simulation. The level of detail should be considered based on the available data and the results desired from the simulation.

(3) **Model implementation.** If the simulation is to be performed using a computer, it must be implemented into the chosen host application. Often the model formulation and implementation effort are carried out simultaneously because the characteristics of the host application may influence how the model is to be constructed.

(4) **Validation of the model.** A model is intended to represent reality. In developing the model, assumptions are made and irrelevant information is omitted. As a result, the model validation is performed to evaluate if the model behaves as expected. There are many techniques available for validation from mathematical analysis to ‘looks right’ analysis. The goal is to increase the confidence that an inference about the real system drawn from the model is accurate.

(5) **Design of the experiment.** Careful thought should be given concerning the number of runs. When random events occur in a simulation, the significance of the results must be considered since the values that have been used in the simulation are sample data used to estimate the parameters of the distribution from which they are drawn. Not enough runs may cause erroneous results while too many runs are unnecessary and can waste time.
(6) **Experimentation.** The simulation model is executed in accordance with the design of experiment to generate the desired data and to perform sensitivity analysis.

(7) **Analysis of results.** Based on the data generated during experimentation, inferences concerning the system of interest can be developed.

(8) **Presentation.** The preceding work is useless unless it provides insight into the system being analyzed. At a minimum, this usually includes presenting the results to a decision-maker. Further work with the model may be required based on a determination from the decision maker.

**TYPES OF SIMULATION**

The aforementioned mathematical model of a sales tax is an example of a deterministic model. That is, the exact outcome can be described by manipulation of the model. Alternatively, the model is probabilistic (or stochastic) if at least one of the variables assumes a range of values over which it can vary and the probabilities with which it will take the values are known. The variable representing the outcome of a random activity is called a stochastic variable.

Thus, a model may be either a physical model or a mathematical model, and may be solved through analytical techniques or through simulation. Additionally, the mathematical model may have deterministic variables or stochastic variables, and may be static or dynamic, depending on whether the variables are time independent or time dependent.

The estimated cost of a system is typically modeled mathematically and, in terms of a point estimate, can be an analytical model. However, given that so many parameters of a typical cost model are uncertain, cost estimates are best studied using stochastic models.

**Deterministic**

Deterministic refers to events that have no random or probabilistic aspects but proceed in a fixed predictable fashion. A deterministic model consists of an exact relationship; for example, if the
labor-rate is known and the man-hours are known, then the cost is known by multiplying the labor rate by the number of man-hours. Unlike a stochastic model, a deterministic equation has no random error. However, there may be error in the variables, meaning labor-rate times man-hours may not be known exactly. In this case, however, there is still no error associated with the equation.

**Stochastic**

Stochastic refers to patterns or processes resulting from random factors. A stochastic model consists of random functions to simulate a deterministic system. Unlike a deterministic model, the equations or their parameters are not known with certainty but only with some amount of probability. For example, the labor rate and the man-hours are only known within some degree of probability and therefore, the cost calculated by labor-rate times man-hours is only known with probability.

**RANDOM NUMBERS**

A random number is one that is drawn from a set of possible values, each of which is equally probable. This is called a uniform distribution, because the distribution of probabilities for each number is uniform (i.e., the same) across the range of possible values. For example, a good (unloaded) die has the probability 1/6 of rolling a one, 1/6 of rolling a two and so on. Hence, the probability of each of the six numbers coming up is exactly the same, so we say any roll of our die has a uniform distribution. Further, each value in a series of random numbers is statistically independent of the preceding values.

A simulation consisting of random components requires a method for generating or obtaining numbers that are random. Most often we generate random numbers from a uniform distribution on the interval of 0 to 1. Random numbers for all other distributions are then obtained by transformation as discussed in the next section.

The random seed is a number that initializes the selection of numbers by a random number generator. Given the same seed, a random number generator will generate the same series of
random numbers each time a simulation is run. Both Crystal Ball and @Risk, by default, pick a
different random seed each time the simulation runs. To avoid this, an initial random seed may be
set. However, if the location of various assumptions is changed on the worksheet, answers will
still vary. ACEIT assigns a random seed to every uncertainty assumption. When the assumption
is moved, the random seed moves with it and therefore the random draw sequence is preserved.
Changing the random seed (either manually or by allowing the tool to do so) can cause the
percentile results to vary on the order of 0.5%. Consequently, it is not possible to get precise
matches across tools since each uses a different random number generator.

**RANDOM VARIATES FROM SPECIFIED DISTRIBUTIONS**

From this random number series, a transformation must be made from the uniformly distributed
sequence of random numbers to the required distribution. A discrete random variable is a mapping
which connects each element of a sample space to particular outcomes. Each value, which the
discrete random variable can take, has a probability associated with each outcome. We write \( P(X = x) \) to mean the probability that the discrete random variable \( X \) takes on the value \( x \). Note \( X \) is
random and \( x \) is a fixed, known value.

Consider one roll of a six-sided die as a discrete random variable. First let’s use this notation:

\[
\begin{align*}
P(X =1) &= 1/6 \\
P(X =2) &= 1/6 \\
P(X =3) &= 1/6 \\
P(X =4) &= 1/6 \\
P(X =5) &= 1/6 \\
P(X =6) &= 1/6
\end{align*}
\]

Where \( X \) = number showing on a die after a roll

Second, the discrete probabilities are summed into a cumulative distribution. Graphic 1 illustrates
how a random number drawn from a 0-1 uniform distribution is “mapped” into discrete outcomes.
In this case a random number draw of \( r=0.49 \) would lie between 2/6 and 3/6 resulting in a simulated
outcome of the die showing 3.
The same scheme could be used to simulate a simple dice game wherein the roller pays $1 when a 1 or 2 is rolled; wins $2 when a six is rolled; and a tie results when a 3, 4, or 5 is rolled. In this case we redefine the discrete random variable to represent the outcome in terms of win/loss amounts with three outcomes notated below and illustrated in Graphic 2:

- \( P(X = -$1) = \frac{2}{6} \)
- \( P(X = 0) = \frac{3}{6} \)
- \( P(X = +$2) = \frac{1}{6} \)

Where \( X = \) player win amount.
This can be easily expanded into a two-die craps example if desired.
The example below illustrates a technique for transforming a random variable sequence.

**Sample Problem**
Tests were conducted on a sample of 50 new rocket assisted artillery projectiles under constant weather
conditions that yielded the following information:

<table>
<thead>
<tr>
<th>Distance (Meters)</th>
<th>Number of Projectiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,000</td>
<td>8</td>
</tr>
<tr>
<td>10,000</td>
<td>16</td>
</tr>
<tr>
<td>11,000</td>
<td>10</td>
</tr>
<tr>
<td>12,000</td>
<td>8</td>
</tr>
<tr>
<td>13,000</td>
<td>4</td>
</tr>
<tr>
<td>14,000</td>
<td>2</td>
</tr>
<tr>
<td>15,000</td>
<td>2</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td><strong>50</strong></td>
</tr>
</tbody>
</table>

This raw data is converted to a histogram which allows the computation of probability for each
event (in this case the distance the projectile travels).

<table>
<thead>
<tr>
<th>Distance</th>
<th>Probability of Occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,000</td>
<td>.16</td>
</tr>
<tr>
<td>10,000</td>
<td>.32</td>
</tr>
<tr>
<td>11,000</td>
<td>.20</td>
</tr>
<tr>
<td>12,000</td>
<td>.16</td>
</tr>
<tr>
<td>13,000</td>
<td>.08</td>
</tr>
<tr>
<td>14,000</td>
<td>.04</td>
</tr>
<tr>
<td>15,000</td>
<td>.04</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td><strong>1.00</strong></td>
</tr>
</tbody>
</table>

Note that each probability has a value between 0 and 1 and the sum of all probabilities = 1.
The probability distribution can be transformed into a cumulative distribution.

<table>
<thead>
<tr>
<th>Distance</th>
<th>Cumulative</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,000</td>
<td>.16</td>
</tr>
<tr>
<td>10,000 or less</td>
<td>.48</td>
</tr>
<tr>
<td>11,000 or less</td>
<td>.68</td>
</tr>
<tr>
<td>12,000 or less</td>
<td>.84</td>
</tr>
<tr>
<td>13,000 or less</td>
<td>.92</td>
</tr>
<tr>
<td>14,000 or less</td>
<td>.96</td>
</tr>
<tr>
<td>15,000 or less</td>
<td>1.00</td>
</tr>
</tbody>
</table>
From the cumulative distribution, a range of random numbers can be constructed and mapping illustrated as shown in Graphic 3.

<table>
<thead>
<tr>
<th>Distance</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,000</td>
<td>.000 - .159</td>
</tr>
<tr>
<td>10,000</td>
<td>.160 - .479</td>
</tr>
<tr>
<td>11,000</td>
<td>.480 - .679</td>
</tr>
<tr>
<td>12,000</td>
<td>.680 - .839</td>
</tr>
<tr>
<td>13,000</td>
<td>.840 - .919</td>
</tr>
<tr>
<td>14,000</td>
<td>.920 - .959</td>
</tr>
<tr>
<td>15,000</td>
<td>.960 - .999</td>
</tr>
</tbody>
</table>

**Graphic 3. Projectile Random Variate**

A simulation of firing the projectile is accomplished by drawing a random number from a table of uniformly distributed random numbers and matching that number with the range values above. This will give the Range in meters that the projectile traveled. For example, if the random number selected is .581, since the random number falls between .480 - .679 this projectile traveled 11,000 meters.
If a sufficient number of random numbers were taken from the table of Uniformly Distributed Random Numbers and matched with the values above, approximately the same distribution observed in the raw data would occur.

The procedure described above is used to transform a sequence of uniformly distributed random numbers into a discrete distribution. The distribution of range, for example, can take on only certain values (9,000; 10,000; etc.). If range were a continuous variable, it could take on any value between 9,000 and 15,000.

SUMMING DISTRIBUTIONS

When WBS elements have been assigned distributions, the sum of those elements is itself a distribution. The mean of a parent WBS element’s distribution is simply the sum of the means of its child WBS elements. However, the output of the primary estimating methodology (i.e. the point estimate) for each WBS item is typically its most likely value. The sum of the most likely values of a number of WBS items is not equal to the most likely value of the WBS sum, except in the unusual case in which the distributions of all the WBS items are symmetric.

The difficulty in summing distribution values other than the mean may be illustrated with a pair of dice. Imagine one die…

- How many possible outcomes can you get from one roll? 6
- How many ways can you roll a six? 1
- How many ways can you roll a number less than or equal to five? 6 – 1 = 5
- What is the likelihood that you will roll a number less than or equal to five? \( \frac{5}{6} = 83\% \)

\[ \rho(x \leq 5) = 0.83 \]
In other words, our 83 percentile confidence level outcome is five or less
If we add two 83 percentile outcomes, do we have an 83 percentile confidence level sum? **NO**…

Imagine two dice…
- How many possible outcomes can you get from one roll? **6 X 6 = 36**
- How many ways can you roll a number greater than ten? **3 (specifically 5-6, 6-5, or 6-6)**
- How many ways can you roll a number less than or equal to ten? **36 – 3 = 33**
- What is the likelihood that you will roll a number less than or equal to ten? **33 / 36 = 92%**

In other words, the sum of two 83 percentile outcomes (two rolls of 5 or less add to 10 or less) is a 92 percentile outcome, not an 83 percentile outcome.

**MONTE CARLO VS. LATIN HYPERCUBE**

Monte Carlo simulation is a method for assessing the overall uncertainty inherent in a model. It calculates the model iteratively using randomly selected values from the error distribution for each of the model components (WBS element CER, input variable, throughputs, etc.), and then using the set of results from all the iterations to estimate the distribution of the overall model results.

The Monte Carlo method has been successfully used in scientific applications for at least 60 years. It is a problem solving technique used to approximate the probability of certain outcomes by running multiple trial runs, called simulations, using random variables. Credit for inventing the Monte Carlo method often goes to Stanislaw Ulam, a Polish born mathematician who worked for John von Neumann on the United States’ Manhattan Project during World War II. Ulam is primarily known for designing the hydrogen bomb with Edward Teller in 1951. He invented the
Monte Carlo method in 1946 while pondering the probabilities of winning a card game of solitaire. Ulam and Metropolis published the first paper on the Monte Carlo method in 1949.

Latin Hypercube sampling (also known as stratified sampling) has been shown to require fewer model iterations to approximate the desired variable distribution than the Monte Carlo method. The Latin Hypercube technique ensures that the entire range of each variable is sampled. It works as follows:

- The distribution is divided into segments of equal probability
- The distribution of samples over these segments is proportional to the probabilities of falling in the segments
- Each sample is drawn from its segment by uniform random sampling

In Graphic 4, a triangular distribution function has been divided into 10 intervals of equal probability (i.e. the area of each interval is the same) and a sample is randomly selected from each interval. Once a sample is taken from a particular interval, this interval is not sampled again in the sampling process until all segments have been sampled. Samples derived by Latin Hypercube method can more accurately reflect the original input distribution with fewer samples when compared to the clustered samples drawn using the Monte Carlo method.
ADDRESSING CORRELATION

Correlation is the term used to describe the degree to which variables are related or associated. An important consideration in risk analysis is to adequately account for the relationships between the cost elements during a risk simulation. This interrelationship between the WBS elements is commonly known as "dependency" or "correlation." For example, if something causes the cost of WBS element A to increase, it may be appropriate that the cost of WBS element B also increases (positive correlation), and perhaps the cost of WBS element F should decrease (negative correlation).

Note that correlation does not change the cost distribution at the lowest levels of the WBS, where uncertainty and correlations are defined. It does however have a significant impact on the uncertainty at the parent levels of the WBS. The specification of positive correlation within an uncertainty assessment will magnify the uncertainty impact at the aggregate level as child WBS elements are forced to move “together.” In contrast, the specification of negative correlation will reduce the uncertainty at parent levels as child WBS elements move in opposing directions.

A very important part of a cost risk uncertainty analysis is an assessment of correlation. If correlation is ignored, the variance at the total levels in the estimate will be understated, in most cases dramatically because, in general, cost uncertainty distributions tend to be positively correlated. If the child element uncertainty distributions are left to be sampled independently of one another, then the high sample on one distribution can be canceled by a simultaneous low one on another element and they will tend to cancel each other out to some extent. For this reason, an estimate without correlation can exhibit too narrow a range. Positive correlation causes elements to move in the same direction (tending to magnify the risk effect), while negative correlation causes elements to move opposite to each other (tending to cancel each other). Implementing positive correlation across the uncertainty estimates will result in broader dispersion (increased variance) of the uncertainty result at the aggregate or parent levels in the WBS.

We are almost ready to demonstrate how to conduct a Monte Carlo simulation using some simple examples and some common tools. But first, we should expand upon steps 7 & 8 (Analysis of
Results and Presentation) of the model building process that were briefly introduced earlier in this lesson. While you are carrying out the first 6 steps, you should already be thinking about interpreting and communicating the results. Many of the software tools available today make it fairly easy to learn how to carry out a basic cost risk uncertainty analysis (e.g., using @Risk to perform a Monte Carlo simulation). Even if you’ve done a superb job of capturing the uncertainty associated with elements within your estimate, you’ve implemented a correlation scheme that you can easily defend, you are familiar with how to implement and run your model within the @Risk (or whatever software tool you are using) and so on...you still have very important tasks ahead. In fact, an analyst’s ability to interpret results and communicate effectively to decision makers may be the most important part of the process.

As a cost analyst, you will ultimately have to present your analysis and recommendations to management—to communicate the results, to get approval, to suggest areas where management should focus attention on the program etc. You have to decide how to communicate your analysis, which depends on the audience, and the purpose of the analysis. Reporting to senior leadership does not typically require the type of detailed charts shown to colleagues or technical management. The key point is to include those items that have meaning to the decision maker and determine how you will portray them (bullets, graphics etc.). Below are some items (this is not an exhaustive list) for you to think about prior to and during your analysis:

- Purpose of the analysis
- Contents of the point estimate
- Assumptions and limitations
- Time-phased TY estimate by appropriation
- General approach of how the uncertainty was defined
- S-Curve (reporting point estimate, mean, 50% and 80% CLs etc.)
- Correlation approach and impacts
- Sensitivity Analysis
- Cost drivers
- Risk mitigation initiatives captured by the estimate
- Schedule drivers
- Recommendations

Again, depending on the audience, you may have to go in to some detail, for example, to explain what the S-curve represents. Does your audience understand correlation and why it’s important?
What does management need to know and where do they need to focus attention? Additionally, what do you do when your analysis suggests that funding/schedule is inadequate? If that is the case, you would need to talk to Program Management personnel. As a result of your analysis and insight, management may decide on the need to:

- Look for cost/schedule/technical trades—(may require discussions/approvals from the user community)
- De-scope the effort
- Request more funding/schedule
- Establish an over target baseline/schedule
- Accelerate or stretch a portion of the program

All of these are management decisions, but you, as a cost analyst may be asked to help provide the analysis/impacts of one of more of these options. See the Joint Agency Cost Schedule Risk & Uncertainty Handbook p. 74 for more on reporting to decision makers.
COMPUTATIONAL CHOICES

The preceding illustrated the essential components of assembling a simulation for a cost estimate. Consider the point estimate model:
Development Cost = Hardware Cost + Software Cost + Test Cost

If the value for each of the three parameters is known with certainty,
Hardware Cost = $100M
Software Cost = $300M
Test Cost = $200M
then the estimate is a simple matter:
Development Cost = Hardware Cost + Software Cost + Test Cost
Development Cost = 100 + 300 + 200
Development Cost = $600M

However, if these values are not certain for the upcoming project, then the project’s cost can only be understood in terms of its total uncertainty, which can be understood via a simulation. Let’s assume the project has determined the following uncertainties. For the purposes of illustrating the tools in this section, the end-points are treated as absolute. Further, these three parameters are correlated 90% with one another.

<table>
<thead>
<tr>
<th></th>
<th>Distribution Shape</th>
<th>Distribution Parameters</th>
<th>Point Estimate</th>
</tr>
</thead>
</table>
| Hardware Cost (SM)   | Triangular         | Minimum = 80
                       |                    | Maximum = 150
                       |                    | Most Likely = 100 | 100             |
| Software Cost (SM)   | Lognormal          | Mean = 300
                       |                    | Standard Deviation = 150 | 300           |
| Test Cost (SM)       | Uniform            | Minimum = 175
                       |                    | Maximum = 275 | 200             |

Fortunately, there are many applications that are available for use in building cost estimates and their simulations. Commercial applications are offered as Excel add-ins. Crystal Ball and @Risk are two popular in use today. A Government application for cost modeling is ACEIT and it
includes a RISK module to perform risk analysis. The following section illustrates the use of the @Risk application by modeling a simple program that includes only 3 elements; Hardware, Software and Testing. Finally we will demonstrate the simulation capability of Excel without the use of any additional add-ins to perform a cost risk/uncertainty analysis of a simple estimate.

@Risk

Building a simulation using @Risk - @RISK is an Excel add-in for the purposes of conducting uncertainty simulations. Open a new Excel workbook, run the @RISK add-in, and enter column titles and values into cells A1 through D5 as shown in Figure 1.

![Figure 1. Point Estimate](image)

To use @RISK, one defines distributions to the cells containing values that are uncertain. In our example we want to define a distribution for each of cells D3, D4, and D5. Place your cursor on cell D3 and select the Define Distributions icon from the menu. The default dialog box as shown in Figure 2 will pop up and we want to define a triangular distribution of 80-150 with a most likely of 100 as per our example. Note the default box has popped up with the Beta General distribution as the default. Select the Triangular distribution by double clicking on “Triang.” Note the input boxes for the default parameters. Enter 80 in Minimum and 150 in Maximum. The result should be as shown in Figure 3. Ignore the remaining items on the dialog box now press OK.
The contents of cell D3 have changed as can be seen in Figure 4. The contents of the cell is now 
“=RiskTriang(80, 100, 150)” replacing our point estimate of 100.
Figure 4. @RISK Function

Repeat the process for cells D4 and D5. Your dialog boxes should appear as shown in Figure 5 and Figure 6.

Figure 5. Define Lognormal Distribution
We next want to specify cell D2 as our simulation output. Highlight that cell and select Add Output icon from the menu. @RISK attempts to name the output based on adjacent cells (Figure 7) but you may enter a name of your choosing. Press OK and the spreadsheet now appear as in Figure 8.
We are now complete with defining the basic simulation and are now ready to specify correlation. Select Define Correlations icon from the menu and an empty matrix will be visible as shown in Figure 9.
To populate this matrix, select the Add Inputs icon and select the cells containing input distributions to be added to the Correlation Matrix (See Figure 10).

![Figure 10. Input Distribution Cells](image)

Select “OK” and a blank Correlation Matrix will appear. Enter 0.9 in each cell (input distribution pairs) to be correlated except the diagonal elements showing 1.000. The resulting correlation matrix should appear as in 11a.

![Figure 11a. Completed Correlation Matrix](image)
Now select “OK” and you will see the window in Figure 11b. Make sure you select a location to place the matrix that DOES NOT over write any of your model as it is very difficult to move.

Figure 11b. Correlation Matrix Location

Once the correlation matrix location is selected, it will be placed on your spreadsheet and look like Figure 11c. (The cells have been changed to “wrap text” to fit all words.) Correlation values can now be changed in the matrix on the spreadsheet.

Figure 11c. Correlation Matrix

Next, open the Model Window icon and verify the information contained on all three tabs: Inputs; Outputs; Correlation. If you did not select an output cell, do that now. Figure 12 shows the Inputs tab of the Model Window.
We now need to configure the simulation settings. Select the small, right most icon under the word “Simulations” in the area marked Simulation from the menu. From this dialog box, we can set several features of the simulation but for our purposes of illustration, we will accept all default settings under the View, Macros and Convergence tabs. We need to set the number of iterations under the General tab which we will change to 500 as shown in Figure 13a.

Select the Sampling tab and you can set the Sampling Type, random number Generator type and Seed Values. See Figure 13b. Select “OK” to close Simulations window.
We are now ready to run the simulation. Select the “Start Simulation” icon and the model will run for the number of iterations you selected. If you have the cursor on the output cell (D2), the default output chart as shown in Figure 14a will appear.
Graphical displays can be changed by selecting different icons across the bottom of Figure 14a. Figure 14b shows the cumulative curve (S-Curve) display.

We can browse through various results charts and tables by selecting icons in the area called “Results.” You can also select from a menu of reports to be displayed in Excel. To do this, select the Excel Reports icon and select Quick Report as shown in Figure 14c. The Excel Reports will be displayed in a new Excel Book. For the Quick Report, three graphs and three data sets are displayed in a single page document. Other reports can be selected and saved in different Excel books.
Spreadsheet

Excel has many features that can be used to construct a simulation without use of a commercial add-in or a specialized tool. However, there are limitations to doing so. First, Excel offers only a few distributions as part of its random number generator drop down, namely lacking Triangular, Lognormal, and Beta distributions. And second, there is no readily available means to perform correlation. For illustrating simulation in Excel, we will simplify our example:

<table>
<thead>
<tr>
<th>Distribution Shape</th>
<th>Distribution Parameters</th>
<th>Point Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Cost (SM)</td>
<td>Normal</td>
<td>Mean = 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Standard Deviation = 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>Software Cost (SM)</td>
<td>Normal</td>
<td>Mean = 300</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Standard Deviation = 80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>300</td>
</tr>
<tr>
<td>Test Cost (SM)</td>
<td>Uniform</td>
<td>Minimum = 175</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maximum = 275</td>
</tr>
<tr>
<td></td>
<td></td>
<td>200</td>
</tr>
</tbody>
</table>

Open a new Excel workbook and enter column titles in cells A1 through E1 as shown in Figure 37. In cells A2 through A201 enter the sequence of numbers from 1 to 200. (Note rows 5-198 are hidden for brevity of the image.)

![Excel Spreadsheet](image.jpg)

*Figure 37. Structure*
From the Excel menu, select Tools | Data Analysis | Random Number Generation. In the Random number Generator dialog, enter 1 Variable and 200 Random Numbers. Select “Normal” from the Distribution drop-down. Ignore the Random Seed field. Select Output Range as the Output Option and enter B2:B201 as the output range. See Figure 38. Press OK and Excel will put a series of random numbers normally distributed around 100 for 200 cells beginning with cell B2 downward. The results should appear as in Figure 39 though exact values will differ since these are randomly generated.

![Random Number Generation dialog box](image)

*Figure 38. Random Number Generator (1 of 3)*
Repeat the sequence of steps for the other two random variates using Figure 40 and Figure 41 for reference. Columns B, C, and D should appear as in Figure 42. Note: since Excel produces these as random numbers, students will not have the exact values as shown on these figures.
In cell E2, enter the cost estimating formula “=SUM(B2:D2)” as shown in Figure 42 and copy the cell into cells E3 to E201.

At this point we have a complete simulation of 200 draws (at least as far as computations go). But, we still need to make interpretation of the results. Excel’s descriptive statistics tool will help us
do that. From the menu, select Tools | Data Analysis | Descriptive Statistics and complete the dialog box as shown in Figure 43. The results are then as shown in Figure 44.

![Descriptive Statistics Dialog](image)

**Figure 43. Descriptive Statistics Dialog**
From these statistics, we can draw various conclusions about the results such as mean, minimum, and maximum. Dividing the standard deviation by the mean can obtain the coefficient of variation 0.132 (81.93806 / 620.3172). In addition to these statistics, it may be valuable to know the shape of the output distribution. The dialog box shown in Figure 45 demonstrates how to use Excel’s Histogram tool, which is accessed through Tools | Data Analysis | Histogram. This tool was used to sort, count into “bins,” and graph the simulation output (200 Development costs), with the resulting histogram and cumulative percentage line shown in Figure 46. If necessary, goodness-of-fit tests could be performed to establish the best fitting output distribution.

![Figure 44. Descriptive Statistics Results](image-url)
Figure 45. Histogram Dialog Box

Figure 46. Histogram
LIMITATIONS OF COMPUTATIONAL CHOICES

There are several tools available that will get the job done. Each has its own advantages and disadvantages. Excel had the limitation in choice of distributions, whereas @Risk (and other tools) offer a much wider choice of distributions. @RISK offers very robust output reports. Interpreting the results using Excel-only required more steps than many other tools require. @RISK obscured the point estimate in the spreadsheet. Excel does not readily permit one to add correlation.
Crystal Ball (for reference only—the screenshots may be out of date for Crystal Ball and ACE-IT).

Crystal Ball is an Excel add-in for the purposes of conducting uncertainty simulations. Open a new Excel workbook, run the Crystal Ball add-in, and enter column titles and values into cells A1 through D5 as shown in Figure 15a.

![Figure 15a. Point Estimate](image)

To use Crystal Ball, one defines distributions to the cells containing the values that are uncertain. In our example, we want to define a distribution for each of the cells D3, D4, and D5. Place your cursor on cell D3 and select Define | Assumption (or icon ![Assumption](image)) and the Distribution Gallery (Figure 15b) is presented. Select Triangular from the available choices and press OK. You are now presented with the Define Assumption dialog. Enter 80 into the Minimum box and 150 into the maximum box and press Enter. The dialog should appear as in Figure 16. Pressing Enter redrew the graph of the distribution. Press OK.
Figure 15b. Distribution Gallery

Triangular Distribution:
The triangular distribution is commonly used when you know the minimum, maximum, and most likely values. For example, you could describe the number of cars sold per week when past sales show the minimum, maximum, and most likely number of cars sold. It is a continuous probability distribution.

The parameters for the triangular distribution are minimum, likelihood, and maximum.

There are three conditions underlying the triangular distribution:
1) The minimum value is fixed.
The content of cell D3 has not changed, but it is now shaded green indicating it has been assigned distribution assumptions. Repeat the process for cells D4 and D5. The Define Assumption dialog for each are shown in Figure 17 and Figure 18.

![Lognormal Distribution](image.png)

*Figure 17. Lognormal Distribution*
We next want to specify cell D2 as our simulation output. Highlight that cell and select Define | Forecast (or select icon ). Using the Define Forecast dialog box, we will determine which reports we want and where Crystal Ball is to write them. Choose tab Auto Extract from within the dialog box and check the box for “Extract forecast statistics automatically to your spreadsheet when the simulation stops”. List a Starting Cell for the output where you have no other data or formulas such as D12. Check every box in the list of statistics as shown in Figure 19. As you check the last one for percentiles an additional box will open as in Figure 20. Select Icosatiles (5% increments) and press OK to return to the define assumption box. Press OK on it and Cell E2 will now be shaded blue to indicate it is a forecast (output) cell.

Figure 18. Uniform Distribution
Figure 19. Define Forecast

Figure 20. Set Percentiles
We are now complete with defining the basic simulation and are now ready to specify correlation. Select Run | Tools | Correlation Matrix from the menu. This will display the Correlation Matrix dialog box as shown in Figure 21.

![Correlation Matrix](image1)

*Figure 22. Correlation Matrix (1 of 3)*

Select each of the three assumptions and press the >> button until each is in the Selected Assumptions list as shown in Figure 22. Press Next.
When step 2 of the Correlation Matrix appears, select the option to Create and link to a correlation matrix on the active workbook and specify F7 as its location. See Figure 23. Press Start.
A blank correlation matrix is placed in the spreadsheet. Enter 0.9 in each of the three empty cells (H8, I8, and I9 in this example) as shown in Figure 24.

![Completed Correlation Matrix](image-url)
We now need to configure the simulation settings. Select Run | Run Preferences to get a Run Preferences dialog. Change the number of trials to 2000 as shown in Figure 25.

![Run Preferences dialog]

**Figure 25. Run Preferences**

The simulation is now ready to run. Select Run | Start Simulation. Two windows will pop up. One shows the simulation progress and the second shows a histogram of the forecast cell(s) being populated as the simulation runs. When complete, the histogram should appear as in Figure 26 and the spreadsheet will now have the forecast cell statistics on it as in Figure 27.
Figure 26. Results Histogram
Figure 27. Results Report
ACEIT

ACEIT is a stand-alone cost modeling application which includes the capability of conducting uncertainty simulations. Open ACEIT and start a new “session” by selecting New from the File menu. For the purposes of this illustration, select OK accepting all defaults. Press F8 to navigate to the Methodology workscreen. Delete all rows that ACEIT may automatically provide. Enter the program point estimate as was done in the @RISK and Crystal Ball examples into rows 1 through 4. Enter the elements names into the WBS/CES Description column. Highlight rows 2 through 4 and press Edit | WBS | Indent to establish parent/child hierarchy. Enter a “C” into each child element of the Phasing Method column. Enter the point estimate values into each child element of the Equation/Throughput column. Press F9 to calculate. Figure 28 presents the point estimate complete in ACEIT.

![ACEIT Methodology](image)

**Figure 28. ACEIT Point Estimate**

To specify the risk distributions, use the Input All dialog box. Put the cursor on row 2 and select View | Input All Form and from within the Input All dialog, select the RISK tab. Use the Distribution drop-down to select “Triangular”. Under the Available Parameters list, select Low and press the >> button to add it to the RISK Specification list. Repeat for High. Note that Low Percentile and High Percentile were also added to the list. In the RISK Specification table, enter 80 for Low and 150 for High. For each, also select the “Val” button because we are entering values as opposed to percentages. Enter 0 for the Low Percentile and 100 for the High Percentile because
we are treating these bounds as absolute. The completed dialog box should appear as in Figure 29.

![Image of Risk Input Dialog for Triangular Distribution]

Figure 29. Risk Input Dialog for Triangular Distribution

Similarly complete the risk inputs for Software and Test Cost elements. For the Lognormal distribution the point estimate is the median, so only the standard deviation need be specified. See Figure 30. For the Uniform, select Undefined for the P.E. Position and then define Low and High as was done for the Triangular. See Figure 31.
Figure 30. Risk Input Dialog for Lognormal
We are now complete with defining the basic simulation and are now ready to specify correlation. In ACEIT, correlation is specified in the Input All dialog box. Put the cursor on row 2 and select View | Input All Form and from within the Input All dialog, select the RISK tab. In the Grouping area in the lower portion of the dialog box, enter an ID of your choosing (in this example we are using “DevItems”) and assign the strength of 0.90. See Figure 32. Repeat for all three items.
Figure 32. Correlation Inputs

Settings for the simulation are set by selecting File | Properties and then selecting the RISK tab on the Properties Dialog Box. Here, set the number of iterations to 500 as shown in Figure 33.
Figure 33. RISK Properties

Now, the simulation is ready to run. Select Calc | Calc with Risk to perform the simulation. Upon completion, the appearance of the values in the Baseline column will change as shown in Figure 34. While the values remain the same, a percentage has been appended showing the confidence level of the point estimate.
To view the detailed results, select Reports | Generate and in the dialog box select Risk Statistics form the Report Type drop-down. Highlight “Risk Stats as Cost and Values” from the list of available reports and press the View button. Figure 35 shows the report.

To view the resulting s-curve, one uses the ACEIT POST tool and results are shown in Figure 36.
Figure 36. ACEIT S-curve