NORTHROP GRUMMAN

DEFINING THE FUTURE

A Non-Simulation Based Method for Inducing Pearson's Correlation Between Input Random Variables

ISPA/SCEA National Conference
May/June 2008

Eric R. Druker
Richard L. Coleman
Peter J. Braxton
Joel B. Hughes

Northrop Grumman Corporation

Acknowledgements

- Thanks to Dr. Steven Book of MCR for his help in obtaining copies of several papers on correlation modules, without which this paper would not have been complete
- Thanks to John Samberg of Tecolote for conversations that helped in the writing of this paper

Introduction

NORTHROP GRUMMAN

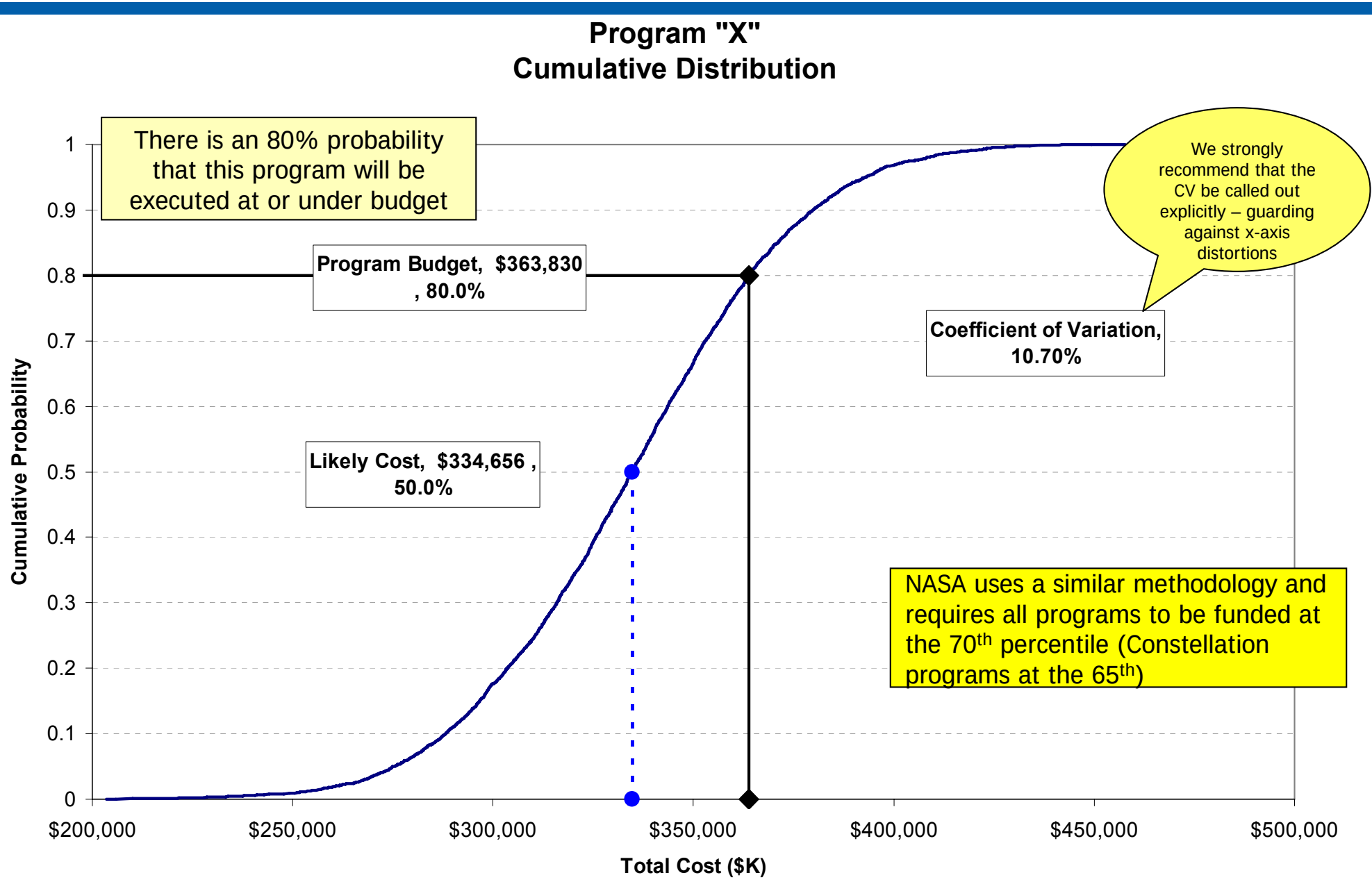
- **Before moving to the main topic of the paper it is important to quickly discuss the motivation behind its development**
- **Studies have shown[1, 2] that 75-85% of DoD programs experience cost overruns**
  - This suggests that as an industry, our estimates are not at the 50th percentile, but rather at about the 20th percentile
- **Recognizing this, agencies are taking the initiative to budget at higher percentiles of cost**
  - NASA requires all programs be funded at the 70th percentile
    - Constellation at the 65th
  - The Air Force (Dr. Sega) has released a memo advising that all space programs be funded at the 80th percentile
    - Rich Hartley (AFCAA) has advised against this, recommending programs be funded at the mean of the AFCAA ICE Estimate (generally between about the 55th and 60th percentiles)
- **In order to determine the appropriate funding level for programs anywhere but at the mean, it is thus imperative that the risk and uncertainty around estimates be assessed**
  - Thus S-Curves must be developed

1 Schaffer 2004 study, referenced from *Cost Estimating Requirements to Support New Congressional Reporting Requirements*. Coonce et. Al. NASA PM Challenge, February 2008

2 NAVAIR Cost Growth Study, R. L. Coleman, M.E. Dameron, C.L. Pullen, J.R. Summerville, D.M. Snead, 34th DoDCAS and ISPA/SCEA 2001

Sample Program S-Curve

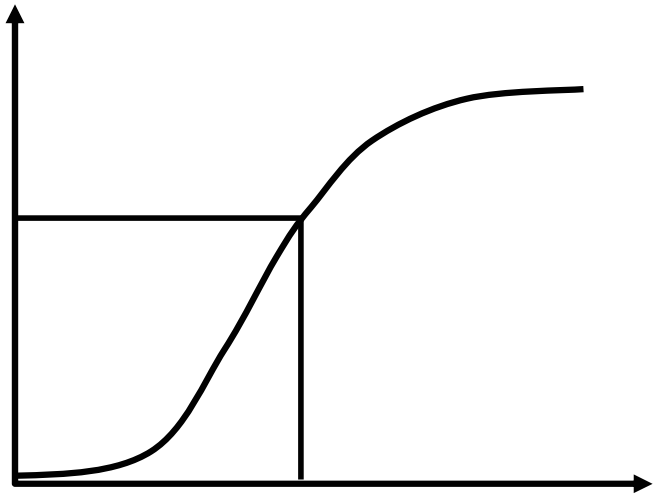
NORTHROP GRUMMAN



S-Curves

NORTHROP GRUMMAN

- **S-Curves are the cumulative distribution function for the cost of a system**
  - Also known as probabilistic cost estimates
- **S-Curves are generally driven by two main factors**
  - Cost Estimating Variance
    - Labor estimates
      - Data Driven
      - SME Driven
    - Escalation/Inflation Rates
    - Material Costs
    - Productivity (e.g. hrs/SLOC, hrs/ft^2)
  - Schedule/Technical Risks and Opportunities
    - Discrete Events
    - Continuous Events



- **Two key measures are derived from these S-Curves**
  - Confidence level of the estimate
    - What is the probability that the program will finish at or under budget?
  - Uncertainty in the estimate
    - What is the range of possibilities for the final cost of this program?
- **The following slides will outline an algorithm that contributes to the development of more accurate S-Curves**

Statement of Problem/Motivation

NORTHROP GRUMMAN

- **Northrop Grumman needed a way to include relational/injected correlation between independent random variables in our risk models**
  - Correlation has a direct effect on the CV of an S-Curve, without it, the CV will be artificially shrunk
  - All models were already capable of functional correlation
- **To determine the best development method, the following factors were considered**
  - NG's risk models need to be easily/electronically transferable via email or network drives
    - Models are shared between a wide range of users in multiple geographic locations
  - The models need to be usable on any machine containing no more than the standard Microsoft Office suite
    - Users often include program/project managers and business management personnel who generally do not have COTS risk tools installed on their computers
    - Users in classified environments have difficulties finding computers with COTS risk tools pre-installed
  - NG must have the ability to implement the method into custom built (generally in Excel) risk models
    - A high percentage or risk analysis work is performed by adding in features to pre-existing cost/price models
- **The previously stated factors drove two decisions:**
  1. A COTS tool would not be used
  2. Visual Basic would be the development platform
     - This would allow the module to have maximum portability
- **In early 2006, work was begun on what would eventually become the "Cost/Risk Correlation Module"**

Development Hurdles

NORTHROP GRUMMAN

- **There were several hurdles that made the module particularly difficult to produce**
  1. The authors had no experience in incorporating relational/injected correlation between independent variables into risk models
  2. There was no single-source authority on how to do so
     - Several papers over the past 20 years have documented various steps, but there was no *one-stop shop* that presented a solution to the problem
  3. Where algorithms did exist they were not plug-and-play
     - Many of the papers consisted only of descriptions of the algorithms
     - Where the full algorithms did exist, they were often in an out-dated programming language
  4. Some of the algorithms, particularly the Lurie-Goldberg algorithm, involve very computationally expensive steps, and a new way of approaching them needed to be developed
     - In particular, the authors could not find any efficient algorithm to find the optimal applied correlation matrix (A') to be described later in the presentation

Outline

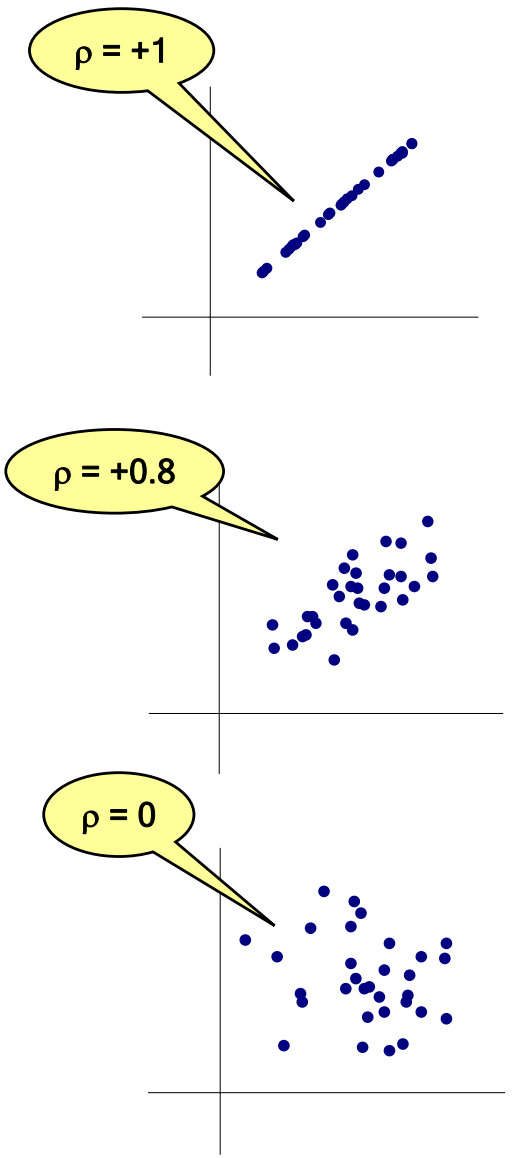
NORTHROP GRUMMAN

- Introduction to Correlation
  - Pearson's "Rho"
  - Pearson's vs. Rank Correlation
- The Problem
- Correlation Matrix Definitions
- Correlation in Risk Models
- Cost/Risk Correlation Algorithm
  - Correcting the user-input matrix
  - Correlating the random variables
  - Optimizing the Applied Matrix

Correlation (Pearson's)

NORTHROP GRUMMAN

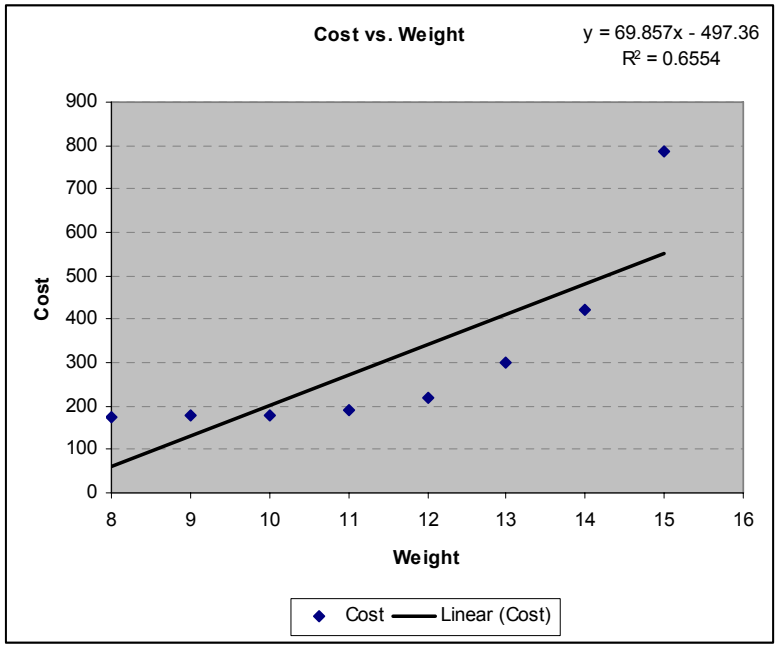
- Although this paper is not about correlation itself, it's important to briefly review the two most common measures
  - Pearson's Product-Moment Correlation
  - Spearman's Rank Correlation
- When correlation is discussed in terms of cost estimating, Pearson's correlation is generally described
- Pearson's Correlation is a measure of the linear relationship between two or more variables
  - This is as opposed to Rank Correlation, which will be discussed on the next slide
- It is identified using the Greek symbol ρ and is always between [-1,1]
- The correlation of a linear regression is the square root of r^2
- The examples on the right show representative data sets for three values of ρ



Pearson's Correlation vs. Rank Correlation

NORTHROP GRUMMAN

- Most commercial risk programs (e.g. Crystal Ball & @Risk) use Spearman's rank correlation rather than Pearson's correlation because it is easier to simulate
- Spearman's rank correlation is used to detect correlation between two variables, without assuming a linear relationship
  - It is concerned with whether or not the function is monotonic
- Some other differences include
  - Pearson's is parametric, Spearman's is not
  - Spearman's is not to be used for predictive purposes
- In the example to the right, rank correlation and Pearson's correlation yield very different answers
- Although it is important to distinguish between these two types of correlation, past research has shown that in cost risk simulations using the most common families of distributions, the two yield fairly similar results[1]
  - The aim of the authors is to "commit no avoidable errors"



Pearson's Rho	**0.81**
Spearman's Rho	**1.00**

[1] Robinson, M and Salls, W. *More on Correlation Accuracy in Crystal Ball Simulations (or What We've Now Learned about Spearman's R in Cost Risk Analyses)*. Presented at the 2004 SCEA Conference, Manhattan Beach, CA, June 2004

Correlation in Risk Models

NORTHROP GRUMMAN

- **In risk analysis, correlations are critical to successful simulations used to find distributions of cost**
  - Correlations are thought to be widely present among elements of cost, but little data exists to determine them, principally because to determine correlations among any set of variables, data points must contain those variables in common, and this is rarely the case
  - Without accounting for correlation, summing multiple independent risk distributions will lead to an artificial degradation in the CV
    - This is known as the "Square Root of N" problem
- **Lacking discernable correlations, risk analysts are forced to rely on Subject Matter Experts to estimate correlations**
  - These correlations are subtle and difficult to estimate
  - Estimated correlations, to be usable, must be "coherent", as discussed later
- **Once the desired correlation between all cost elements is determined, the next problem is to build these correlations into the risk model**
- **The following slides will present a quick method for inducing Pearson's correlation between input random variables while preserving their marginal distributions**

Definitions: Matrices

NORTHROP GRUMMAN

- **Before proceeding, it is important to define several matrices that will be used in the algorithm**
- **Input Correlation Matrix:**
  - The correlation matrix inputted by the user
  - May or may not be a consistent correlation matrix
- **Adjusted Correlation Matrix:**
  - The consistent correlation matrix found by the model that is as close as possible to the Input Correlation Matrix
    - This matrix is positive semidefinite
    - It is also coherent given the distributions being correlated
- **Applied Correlation Matrix:**
  - The correlation matrix utilized by the algorithm to generate correlated random number draws
- **Outcome Correlation Matrix:**
  - The correlation matrix of the simulation variables after the simulation is run
  - Ideally it is identical to the Adjusted Correlation Matrix

User-Input Matrix		
1.0000	0.8000	0.1000
0.8000	1.0000	0.8000
0.1000	0.8000	1.0000

Adjusted Matrix		
1.0000	0.7522	0.1322
0.7522	1.0000	0.7522
0.1322	0.7522	1.0000

Applied Matrix		
1.0000	0.7915	0.2263
0.7915	1.0000	0.7744
0.2263	0.7744	1.0000

Outcome		
1.0000	0.7522	0.1316
0.7522	1.0000	0.7521
0.1316	0.7521	1.0000

Definitions: Eigenvalues/Eigenvectors

NORTHROP GRUMMAN

- An eigenvector is a vector v such that for a square matrix A and a scalar λ, $Av = \lambda v$
- It follows that if Q is an indexed set of linearly independent eigenvectors for matrix A and Λ is the diagonal matrix containing the corresponding eigenvalues of A as its diagonal entries then:

  $A = Q\Lambda Q^{-1}$

- By altering Λ, the diagonal matrix consisting of A's eigenvalues, we eventually arrive at a positive definite correlation matrix that is close to the user input matrix
- The Jacobi Eigenvalue algorithm is used to find both the eigenvalues and eigenvectors of the user input correlation matrix

NORTHROP GRUMMAN

The Cost Risk Correlation Algorithm

Correcting the User Input Matrix

Correlating the Uniform Random Number Draws

Optimizing the Applied Matrix

Correcting the User Input Matrix

NORTHROP GRUMMAN

- **As a rule, correlation matrices must be positive semidefinite**
  - Positive semidefinite matrices have all non-negative eigenvalues
- **When using data to generate correlation matrices, they will necessarily be positive semidefinite**
- **Unfortunately, when generating matrices based on SME judgment, this condition may not be met**
- **To correct these matrices, an algorithm developed by Iman and Davenport[1] was used**
  - The criteria for "closest matrix" that comes out of this algorithm is unknown to the authors but it is computationally efficient and relatively simple to implement
  - Because the generation of the "closest viable correlation matrix" is so critical in finance, there are several more robust algorithms available[2]
- **The following slide will outline the algorithm used in the Cost-Risk Correlation Module**

[1] Iman, R and Davenport J. *An Intterative Algorithm to Produce a Positive Definite Matrix from an "Approximated Correlation Matrix" (With a Program User's Guide)* Sandia National Laboratories for the US DoE, June 1982

[2] Higham, N. *Computing the Nearest Correlation Matrix – A Problem from Finance.* IMA Journal of Numerical Analysis. 2002

Correcting the User Input Matrix - Hurdles

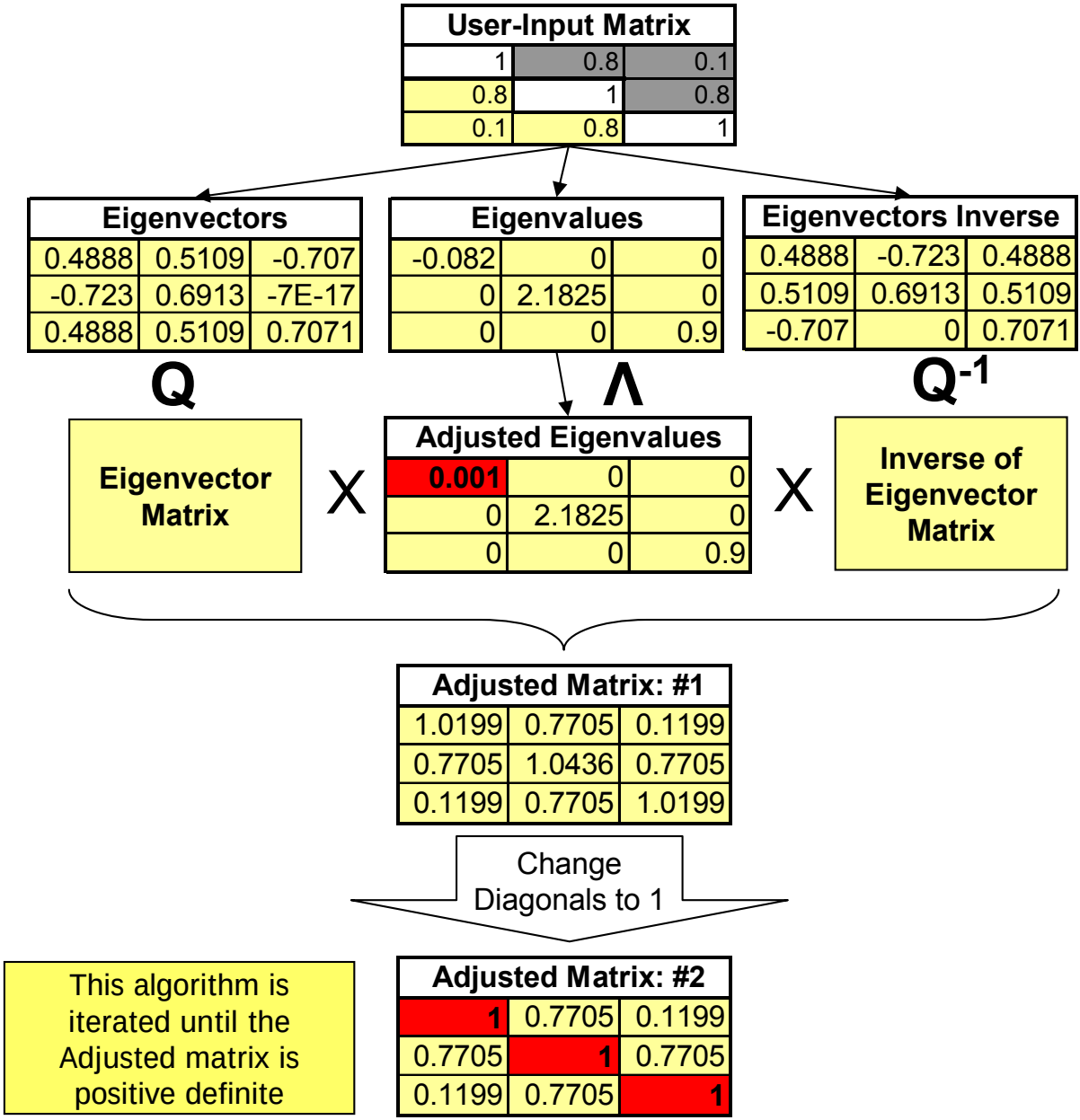
NORTHROP GRUMMAN

- **Two hurdles existed in implementing the algorithm**
  - Excel doesn't have a function that finds Eigenvalues and Eigenvectors for the correlation matrices
  - Excel doesn't have a function to compute the Cholesky Decomposition matrix
- **Research was conducted and algorithms (and the associated VBA source code) that conquered both hurdles were found**
  - Both were part of the MATRIX and LINEAR ALGEBRA Package For EXCEL developed by The Foxes team in Italy
  - The Cholesky Decomposition, Eigenvalues and Eigenvectors functions were taken from this package and added into the tool

Correcting the User Input Matrix - Algorithm

NORTHROP GRUMMAN

- The algorithm iteratively adjusts the eigenvalues of user-inputted correlation matrices until the resulting matrix has all positive eigenvalues
- During each iteration of the algorithm, there are two adjustments
  1. Adjustment of the negative eigenvalues to small, positive values
  2. Adjustment of the first adjusted matrix's diagonal entities to values of 1
- Once the adjusted matrix (#2) is found to have all non-negative Eigenvalues, the algorithm has found its solution



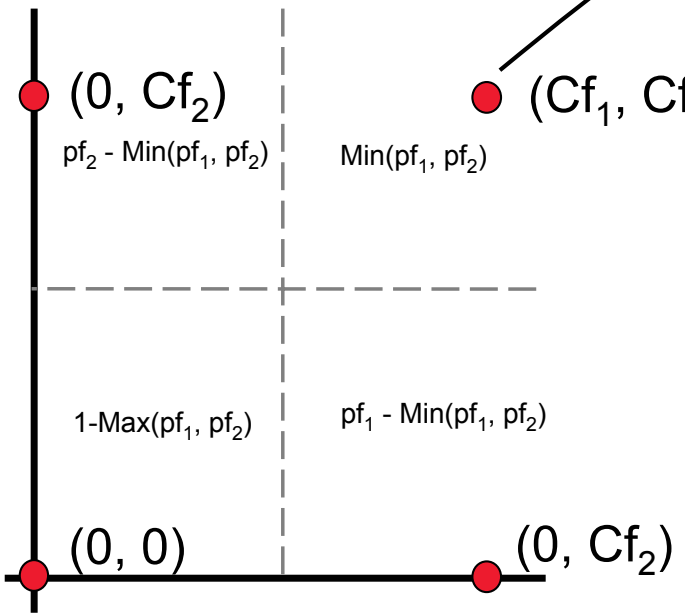
Correcting the User Input Matrix – Other Complications

NORTHROP GRUMMAN

- Although the matrix produced using the algorithm on the preceding slides is a consistent correlation matrix, depending on the random variables being correlated it may or may not be feasible
  - At least if the marginal distributions are to be preserved
- The best way to illustrate this is to examine the maximum possible correlation between two Bernoulli risks
  - As shown below, unless the probabilities of the two risks are equal, there is a maximum possible correlation between them
- The final step to correcting the User Input Matrix is to adjust the matrix so that all correlations are feasible based on the distributions being correlated

Although this example seems odd, this is an efficient way of inducing conditional probabilities between Bernoulli random variables

The only case in which XY ≠ 0 is when both risks occur, it follows that E(XY) simplifies down to $Cf_1 x Cf_2$ times the probability that both risks occur. The highest this probability can possibly be is the minimum of the two probabilities



$$\rho_{X,Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}}$$

$$\rho(\max)_{X,Y} = \frac{Min(Pf_1, Pf_2) \times Cf_1 \times Cf_2 - (Pf_1 \times Cf_1) \times (Pf_2 \times Cf_2)}{\sigma_X \times \sigma_Y}$$

$$\rho(\max)_{X,Y} = \frac{Min(Pf_1, Pf_2) \times Cf_1 \times Cf_2 - (Pf_1 \times Cf_1) \times (Pf_2 \times Cf_2)}{Cf_1\sqrt{Pf_1 \times Qf_1} \times Cf_2\sqrt{Pf_2 \times Qf_2}}$$

$$\rho(\max)_{X,Y} = \frac{Min(Pf_1, Pf_2) - Pf_1 \times Pf_2}{\sqrt{Pf_1 \times Qf_1}\sqrt{Pf_2 \times Qf_2}}$$

Correlating Random Variables: An Introduction to the Lurie-Goldberg Method[1]

NORTHROP GRUMMAN

- The only method the authors were aware of for inducing Pearson's correlation between input random variables is the Lurie-Goldberg Algorithm
  - The Lurie-Goldberg Algorithm aims to find an applied correlation matrix such that the input correlation and output correlation are as close as possible

- Find matrix **L** such that series of transformations

$$X \xrightarrow{\mathbf{L}} Y \xrightarrow{\Phi} U \xrightarrow{F^{-1}} V$$

indep. normal — *mult. normal* — *uniform* — *desired*

lead to random variables with desired correlations and marginal distributions

- **L**: Cholesky factor transforms independent normals to correlated normals
- Φ: normal c.d.f. transforms correlated normals to correlated uniforms
- F^{-1}: transforms correlated uniforms to correlated random variables with desired marginal distributions F

  - Unfortunately, the authors could not find a method for finding this optimal matrix (L… referenced as A' in this paper)
  - One obvious solution is to optimize the matrix by examining the post-simulation correlations
    - Given the computing power needed to complete each simulation, this could be a time consuming endeavor

[1]Goldberg, Matthew S, Lurie, Phillip M. *Correlating Random Variables*, 32nd DoDCAS, Williamsburg, VA. February 1999

Correlating the Uniform Draws: The Lurie-Goldberg Method

NORTHROP GRUMMAN

- Once a viable correlation matrix exists Uniform (0,1) correlated random numbers must be generated which in turn are used to generate the desired random variables
- To accomplish this, the Cholesky Decomposition Matrix of the adjusted matrix is found
  - L is the Cholesky Decomposition of A iff L is a lower triangular matrix such that:

$$A = LL^T$$

- After the Cholesky Decomposition Matrix is found, the algorithm at right is run to produce correlated Uniform (0,1) random numbers
- These random numbers, vice the originals, are used in the risk model to generate points off of distributions

Adjusted Matrix		
1.0000	0.7522	0.1322
0.7522	1.0000	0.7522
0.1322	0.7522	1.0000

Cholesky Decomposition

Cholesky Decomposition		
1.0000	0.0000	0.0000
0.7522	0.6589	0.0000
0.1322	0.9907	0.0321

X

U(0,1) Random Draws
0.26271853333989800
0.79616660202169400
0.15362541632109700

Inverse CDF Technique

Random N(0,1)
(0.63498673467686800)
0.82800654029771300
(1.02100761130346000)

Multiply N(0,1) by Cholesky

Correlated Random N(0,1)
(0.63498673467686800)
0.06794090908429620
0.70360627328862900

Multiply N(0,1) by Cholesky

Correlated Random U(0,1)
0.26271853333989800
0.52708366338494000
0.75916099823068700

Note: The resulting correlation between the Correlated Random U(0,1) random numbers will not be exactly the same as the adjusted correlation matrix… **more on this soon**

Outcome Correlation		
1.0000	0.7386	0.1367
0.7386	1.0000	0.7395
0.1367	0.7395	1.0000

Resulting Correlation

Optimizing the Applied Correlation Matrix

NORTHROP GRUMMAN

- **Non-linear transformations are used to correlate random variables in the model**
  - Because of this, the outcome correlation may be different from the intended correlation
- **The biggest hurdle this module faced was in the correction of this discrepancy**
- **Northrop Grumman has developed a method that can find the outcome correlation matrix for any applied correlation matrix prior to the simulation being run**
  - In other words, the algorithm can determine ρ_{Output} given $\rho_{Applied}$
  - The applied correlation matrix can then be optimized so that the outcome correlation matrix is equal to the adjusted correlation matrix
- **Additionally, it follows from proofs that the optimal applied correlation matrix will induce the desired correlation**
  - This infers that any variation in ρ in the simulation runs is due solely to Monte Carlo sampling error

Find:

Applied Correlation Matrix		
1.0000	0.7915	0.2263
0.7915	1.0000	0.7744
0.2263	0.7744	1.0000

Such that after the Lurie-Goldberg method takes place:

Outcome Correlation Matrix		
1.0000	0.7522	0.1322
0.7522	1.0000	0.7522
0.1322	0.7522	1.0000

=

Adjusted Correlation Matrix		
1.0000	0.7522	0.1322
0.7522	1.0000	0.7522
0.1322	0.7522	1.0000

Optimizing the Applied Correlation Matrix

NORTHROP GRUMMAN

- **The algorithm developed by Northrop Grumman finds the optimal applied correlation matrix given:**
  1. The parent distributions being correlated
  2. The adjusted correlation matrix
- **The algorithm runs prior to the simulation being executed and once performed, only needs to be re-ran as variables are added or changed**
  - And in those cases, only for the new/modified distributions
- **Although the algorithm was originally developed for cost risk analysis, it has applications wherever a user needs to account for correlation between independent random variables**
  - For example: the modeling of mutual fund performance given it is made up of a group of correlated stocks and bonds
- **In fact, the algorithm's first use is in the modeling of conditional probabilities between Bernoulli independent random variables**
  - The customer needed an efficient way to model the conditional probabilities they found between parameters in their data while preserving the marginal probabilities
  - It can be shown using the same general methodology on slide 14 that Pearson's correlation between two Bernoulli random variables is equivalent to a conditional probability between them