APPENDIX D

STRUCTURAL RELIABILITY ANALYSIS

Development of FORM model in using spreadsheet

In this study, FORM model was built using Microsoft Excel and Visual Basic for Application. A practical procedure presented by Zhao and Ono (1999), Low and Tang (2004) and Low and Tang (2007) was used to develop FORM model in Excel. The reliability index was obtained by calling Excel's built-in optimization program, Solver, with the objective function to minimize the reliability index subject to the constraint that the limit-state function, *g*(*x*) = 0. The objective function is

$$\text{Minimize: } \boldsymbol{\beta} = \mathbf{min}_{\bar{\mathbf{x}} \in \mathbf{F}} \sqrt{\left[\frac{\mathbf{x_i - m_i}}{\boldsymbol{\sigma}_\mathbf{i}}\right]^{\mathbf{T}} [\mathbf{R}]^{-\mathbf{1}} \left[\frac{\mathbf{x_i - m_i}}{\boldsymbol{\sigma}_\mathbf{i}}\right]}$$

where $\bar{\boldsymbol{x}}$ is a vector representing the set of random variables, $\bar{\mathbf{m}}$ is the mean value, $\bar{\mathbf{R}}$ is the correlation matrix and **F** is the failure domain. The probability of failure is then calculated. Figure D1 represents FORM framework for assessing the failure probability.

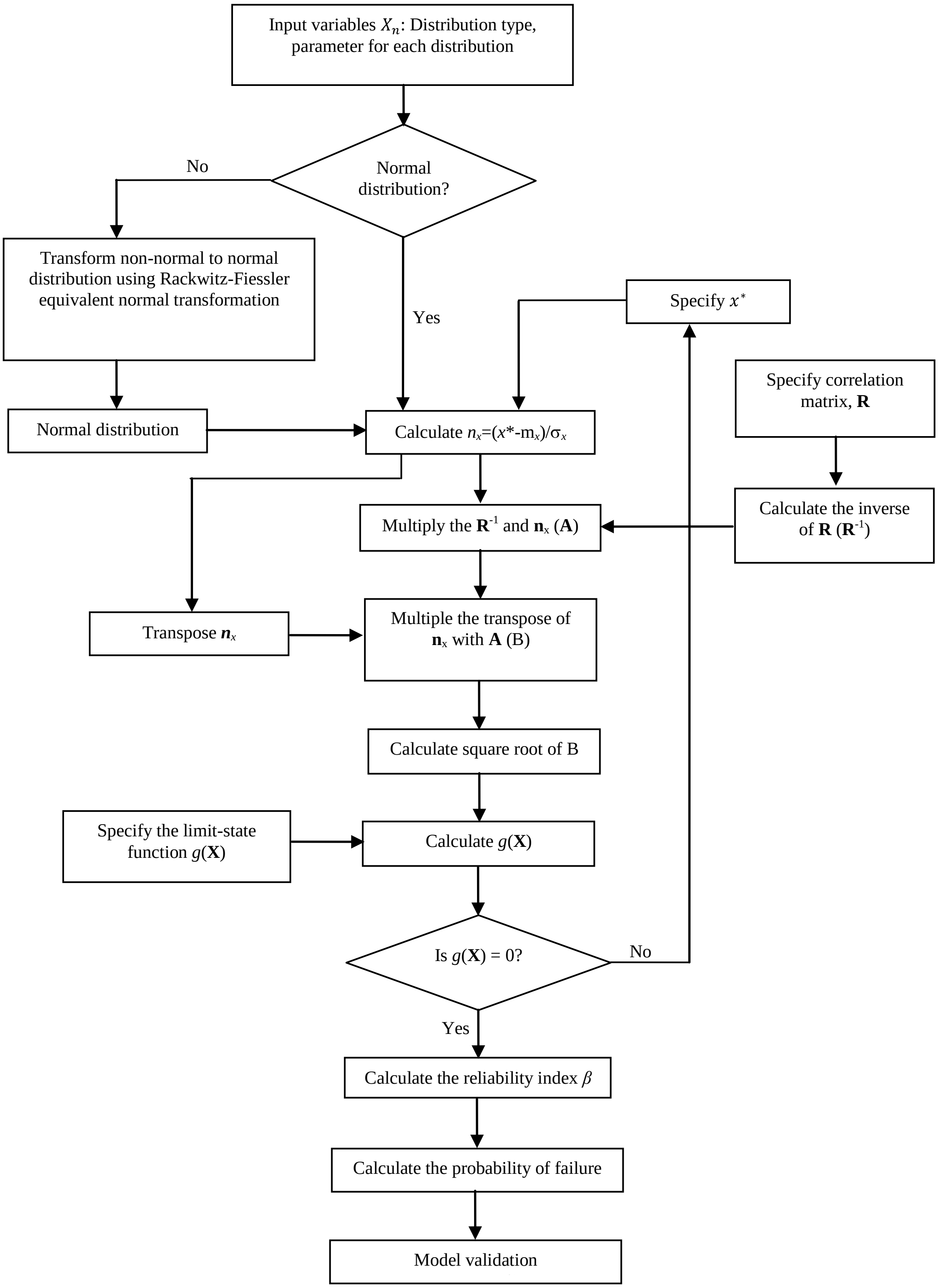

Input variables X_n: Distribution type, parameter for each distribution
Normal distribution?
No
Yes
Transform non-normal to normal distribution using Rackwitz-Fiessler equivalent normal transformation
Normal distribution
Specify x^*
Specify correlation matrix, **R**
Calculate n_x=(x*-m$_x$)/σ$_x$
Multiply the $\mathbf{R}^{-1}$ and $\mathbf{n}_x$ (**A**)
Calculate the inverse of **R** ($\mathbf{R}^{-1}$)
Transpose $\boldsymbol{n}_x$
Multiple the transpose of $\mathbf{n}_x$ with **A** (B)
Calculate square root of B
Specify the limit-state function g(**X**)
Calculate g(**X**)
Is g(**X**) = 0?
No
Yes
Calculate the reliability index β
Calculate the probability of failure
Model validation

Figure D1: FORM Model Framework in Excel

Estimation of probability distribution and its parameter

It is evident that the limit state function used in this study is a function of five basic variables (i.e. material strength S, diameter D, material thickness t, operating pressure P and corrosion rate CR). To conduct FORM analysis, the variability of the random variables has to be estimated in terms of their probability distributions and their parameters:

1. The variability in material property (i.e. material strength S)
   - Material strength is often used to represent the limit state, in structural reliability problems of corroded systems and hence, is crucial information in structural reliability assessments. It is also subjected to variations and is considered as a basic random variable that can be statistically represented.
   - The variation in material property is typically assumed normally distributed for it is a reasonable model for many natural processes or physical properties (Melchers, 1999). Other distributions that have been suggested for representing the distribution of material strength are lognormal and Extreme Value Type I distributions. The mean value of the material strength is assumed to be the value typically specified for specific pipe material under study. COV will be based on the Swedish yield stress data in Melchers (1999, pg. 278) as shown in Table D.1.

Table D.1: Swedish yield stress data (Melchers, 1999)

Nominal yield stress (MPa)	COV	Number of samples
220	0.103	19,857
260	0.099	19,217
360	0.057	11,170
400	0.054	2447

Note: Number of samples is the number of data points to estimate coefficient of variation

2. The variability in pipe properties (i.e. diameter D, material thickness t)
   - Relatively the initial pipe diameter and thickness are not available. Therefore, the variation in pipe diameter and thickness is assumed to follow a normal distribution since many processes are assumed to follow this distribution (Melchers, 1999). The mean value of pipe diameter and thickness is assumed to be the value of the specified diameter and the nominal pipe thickness,

respectively. For COV, Ellingwood et al. (1999) suggested an average COV of 0.05.

- According to ASME SA 530 or ASTM A530 specification, the requirement for pipe wall thickness and diameter tolerance is 12.5%. Assume the pipe wall thickness is 5.54 mm. The tolerance for the pipe is ±0.35 mm which means the thickness has to be manufactured between 5.19 to 5.89 mm. Table D.2 shows how the variation in the manufacturing process can be translated into COV. Figure D2 shows the effect of having different values of COV on the pipe thickness distribution. Therefore, the COV value of 0.05 suggested by Ellingwood et al. (1999) is considered in this study since it is within the tolerance specified by the standard requirement.

Table D.2: Variation in manufacturing process

Manufacturing process	σ	**COV**
σ = 0.35	0.35	0.06
2 σ = 0.35	0.175	0.03
3 σ = 0.35	0.117	0.02
6 σ = 0.35	0.058	0.01

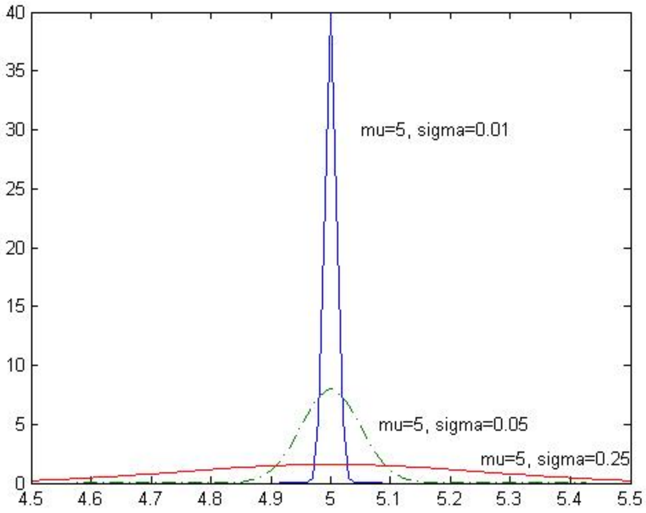


Figure D.2: The effect of having different values of COV on the pipe thickness distribution

3. The variability in operating parameters (i.e. operating pressure P)
   - To estimate the probability distribution for operating pressure and its parameters, the actual historical operating pressure data is taken for respective pipe under study. Then, the data is analyzed to establish its probability distribution and parameters using the distribution fitting method.
4. The variability in corrosion rate CR
   - In structural reliability assessment, the defect growth rate is the primary input to allow future prediction of the defect size, which in turn allows prediction of the failure probability. The traditional corrosion analysis assumes a deterministic defect growth function, even though in reality, the defect growth rates also exhibit a large stochastic behavior. In this study, the probability distribution of CUI corrosion rate is to be established using corrosion rate data collected during inspection period. The method to estimate the probability distribution for corrosion rate and its parameters has been discussed in Chapter 3 and 5 using the bootstrapping method.

Model Validation

Data from Teixeira et al. (2008) was used to validate the FORM spreadsheet model as shown in Table D.3. The limit-state function for intact pipes is given by

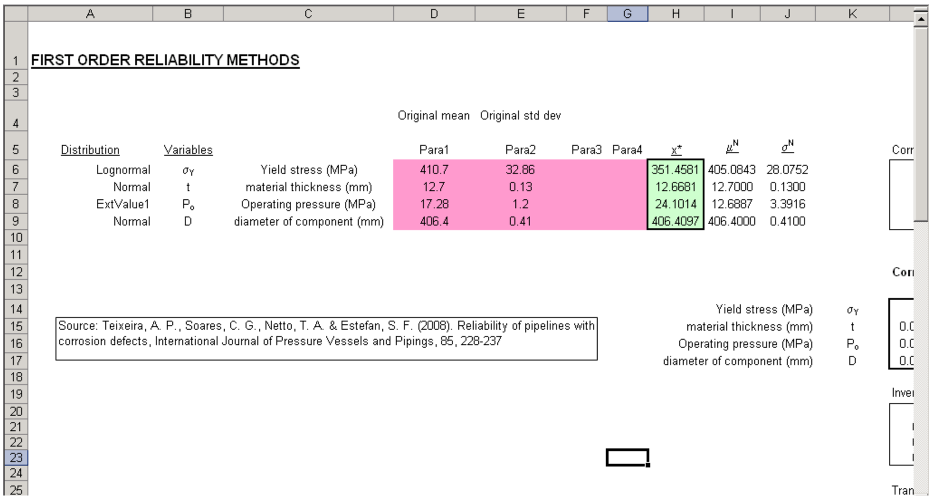
$$g(\mathbf{X}) = \left(\frac{1.1\sigma_y 2t}{D} \right) - P_o$$

where σ_y is the material yield stress, D is the outside diameter of the pipe, t is the wall thickness of the pipe and P_o is the internal operating pressure of the pipe.

Table D.3: Probabilistic models of the basic variables of the intact pipe

Basic variables	Distribution	Mean	Std. Dev.
Yield stress, σ_y	Lognormal	410.7 MPa	32.86 MPa
Diameter, D	Normal	406.4 mm	0.41 mm
Thickness, t	Normal	12.7 mm	0.13 mm
Operating pressure, P_o	Gumbel	17.28 Pa	1.20 mm

Figure D3 and D4 show the Excel template developed and how to run Solver in order to get the solutions.



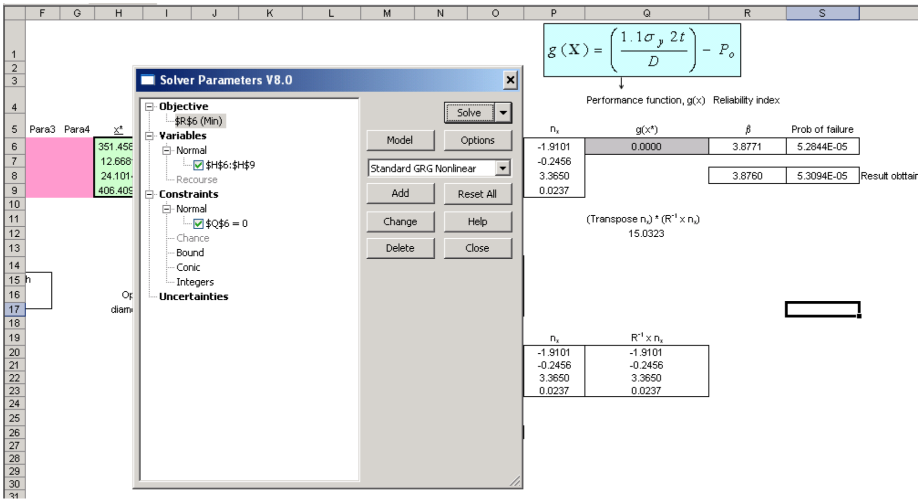
FIRST ORDER RELIABILITY METHODS

Distribution	Variables		Para1	Para2	Para3	Para4	x*	μ^N	σ^N
			Original mean	Original std dev					
Lognormal	σ_Y	Yield stress (MPa)	410.7	32.86			351.4581	405.0843	28.0752
Normal	t	material thickness (mm)	12.7	0.13			12.6681	12.7000	0.1300
ExtValue1	P_o	Operating pressure (MPa)	17.28	1.2			24.1014	12.6887	3.3916
Normal	D	diameter of component (mm)	406.4	0.41			406.4097	406.4000	0.4100

Source: Teixeira, A. P., Soares, C. G., Netto, T. A. & Estefan, S. F. (2008). Reliability of pipelines with corrosion defects, International Journal of Pressure Vessels and Pipings, 85, 228-237

Yield stress (MPa)	σ_Y	0.0
material thickness (mm)	t	0.0
Operating pressure (MPa)	P_o	0.0
diameter of component (mm)	D	0.0

Figure D3: Excel template for FORM model



Solver Parameters V8.0

- Objective
  - $R$6 (Min)
- Variables
  - Normal
    - ☑ $H$6:$H$9
  - Recourse
- Constraints
  - Normal
    - ☑ $Q$6 = 0
  - Chance
  - Bound
  - Conic
  - Integers
- Uncertainties

Standard GRG Nonlinear

$$g(\mathbf{X}) = \left(\frac{1.1\sigma_y\,2t}{D}\right) - P_o$$

Performance function, g(x)

Reliability index

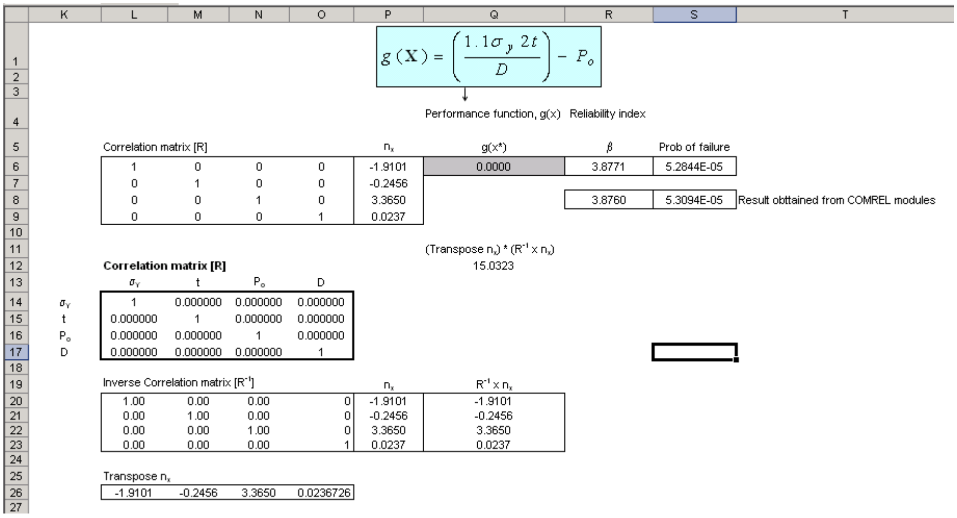
n_x	g(x*)	β	Prob of failure	
-1.9101	0.0000	3.8771	5.2844E-05	
-0.2456				
3.3650		3.8760	5.3094E-05	Result obttair
0.0237				

(Transpose n_x) * (R^{-1} x n_x)
15.0323

n_x	R^{-1} x n_x
-1.9101	-1.9101
-0.2456	-0.2456
3.3650	3.3650
0.0237	0.0237

Figure D4: Solver to solve the limit state function

Figure D5 shows the results generated by FORM model developed and it showed that the result is almost identical as the result published in the paper by Teixeira et al. (2008). Therefore, the developed model is validated.



	K	L	M	N	O	P	Q	R	S	T
1						$g(\mathbf{X}) = \left(\frac{1.1\sigma_y\ 2t}{D}\right) - P_o$				
2										
3							↓			
4							Performance function, g(x)	Reliability index		
5		Correlation matrix [R]				n_x	g(x*)	β	Prob of failure	
6		1	0	0	0	-1.9101	0.0000	3.8771	5.2844E-05	
7		0	1	0	0	-0.2456				
8		0	0	1	0	3.3650		3.8760	5.3094E-05	Result obttained from COMREL modules
9		0	0	0	1	0.0237				
10										
11							(Transpose n_x) * (R^{-1} x n_x)			
12		**Correlation matrix [R]**					15.0323			
13		σ_Y	t	P_o	D					
14	σ_Y	1	0.000000	0.000000	0.000000					
15	t	0.000000	1	0.000000	0.000000					
16	P_o	0.000000	0.000000	1	0.000000					
17	D	0.000000	0.000000	0.000000	1					
18										
19		Inverse Correlation matrix [R^{-1}]				n_x	R^{-1} x n_x			
20		1.00	0.00	0.00	0	-1.9101	-1.9101			
21		0.00	1.00	0.00	0	-0.2456	-0.2456			
22		0.00	0.00	1.00	0	3.3650	3.3650			
23		0.00	0.00	0.00	1	0.0237	0.0237			
24										
25		Transpose n_x								
26		-1.9101	-0.2456	3.3650	0.0236726					
27										

Figure D5: Generated results by FORM model

Excel VBA Coding

Below are parts of the VBA coding used in the FORM model. This coding is used to transform any distributions to normal distribution.

```
Function EqvN(DistributionName, para1, para2, para3, para4, x, code)

'This function is used to transform any types of distributions to normal distribution

'del is the value assign to x if x < del (to avoid division of 0)

del = 0.0001


'Select the following case for specified distribution.
'UCase function returns a Variant (String) containing the specified string,
'converted to uppercase.

Select Case UCase(Trim(DistributionName))

   Case "NORMAL":

      If code = 1 Then EqvN = para1
      If code = 2 Then EqvN = para2

   Case "UNIFORM":

      a1 = para1: a2 = para2: min = para3: max = para4
```

```
    If x <= min Then x = min + del
    If x >= max Then x = max - del

    cdf = Application.BetaDist(x, a1, a2, min, max)
    pdf = Betapdf1(x, a1, a2, min, max)

    EqvN = EqvTransform(x, cdf, pdf, code)

Case "LOGNORMAL":

    para3 = 0
    para4 = 0

    If x < del Then x = del

    lambda = Log(para1) - 0.5 * Log(1 + (para2 / para1) ^ 2)

    If code = 1 Then
        EqvN = x * (1 - Log(x) + lambda)
    ElseIf code = 2 Then
        EqvN = x * Sqr(Log(1 + (para2 / para1) ^ 2))
    End If

Case "EXTVALUE1":

    para3 = 0
    para4 = 0

    alfa = 1.2825498302 / para2
    u = para1 - 0.5772 / alfa

    cdf = Exp(-Exp(-alfa * (x - u)))
    pdf = alfa * Exp(-alfa * (x - u)) * cdf

    EqvN = EqvTransform(x, cdf, pdf, code)

Case "EXPONENTIAL":

    para2 = 0
    para3 = 0
    para4 = 0

    b = para1

    If x < del Then x = del

    cdf = 1 - Exp(-x / b)
    pdf = 1 / b * Exp(-x / b)
    EqvN = EqvTransform(x, cdf, pdf, code)

Case "GAMMA":

    para3 = 0
    para4 = 0

    If x < del Then x = del
    cdf = Application.GammaDist(x, para1, para2, True)
    pdf = Application.GammaDist(x, para1, para2, False)
    EqvN = EqvTransform(x, cdf, pdf, code)

Case "WEIBULL":

    para3 = 0
    para4 = 0
    If x < del Then x = del
    cdf = Application.Weibull(x, para1, para2, True)
    pdf = Application.Weibull(x, para1, para2, False)
    EqvN = EqvTransform(x, cdf, pdf, code)

Case "TRIANGULAR":

    para4 = 0

    a = para1: Mode = para2: c = para3

    If x <= a Then x = a + del
    If x >= c Then x = c - del
    If x < Mode Then cdf = (x - a) ^ 2 / (Mode - a) / (c - a)
    If x < Mode Then pdf = 2 * (x - a) / (Mode - a) / (c - a)
    If x >= Mode Then cdf = 1 - (c - x) ^ 2 / (c - a) / (c - Mode)
    If x >= Mode Then pdf = 2 * (c - x) / (c - a) / (c - Mode)
    EqvN = EqvTransform(x, cdf, pdf, code)


Case "BETADIST":

    a1 = para1: a2 = para2: min = para3: max = para4

    If x <= min Then x = min + del
    If x >= max Then x = max - del
    cdf = Application.BetaDist(x, a1, a2, min, max)
    pdf = Betapdf1(x, a1, a2, min, max)

    EqvN = EqvTransform(x, cdf, pdf, code)
    Case "PERTDIST":

        para4 = 0

        a = para1: Mode = para2: c = para3

        If x <= a Then x = a + del
        If x >= c Then x = c - del

        mean = (a + 4 * Mode + c) / 6


        If Mode = mean Then
            f = 6
        Else
            f = (2 * Mode - a - c) / (Mode - mean)

        End If

        a1 = (mean - a) * f / (c - a)
        a2 = a1 * (c - mean) / (mean - a)

        cdf = Application.BetaDist(x, a1, a2, a, c)

        pdf = Betapdf1(x, a1, a2, a, c)

        EqvN = EqvTransform(x, cdf, pdf, code)

End Select

End Function

Function EqvTransform(x, cdf, pdf, code)

epsi = 10 ^ (-16)

If cdf < epsi Then cdf = epsi
If cdf > 1 - epsi Then cdf = 1 - epsi

EqvSigma = Application.NormDist(Application.NormSInv(cdf), 0, 1, False) / pdf

If code = 1 Then EqvTransform = x - EqvSigma * (Application.NormSInv(cdf))
If code = 2 Then EqvTransform = EqvSigma

End Function

Function betapdf(x, a1, a2, min, max)

With aplication.WorksheetFunction
```

```
BetaFunc = Exp(.GammaLn(a1) + .GammaLn(a2) - .GammaLn(a1 + a2))

End With

    betapdf = 1 / BetaFunc * (x - min) ^ (a1 - 1) * (max - x) ^ (a2 - 1) / (max - min) ^ (a1 + a2 - 1)

End Function

Public Function TEST(x, Y)
TEST = x + Y

End Function

Public Function SQRT(a1)

With aplication.WorksheetFunction

a = .SQRT(a1)

End With

End Function
```