NASA/CR—1999-209154

Structural Reliability Analysis and Optimization: Use of Approximations

Ramana V. Grandhi and Liping Wang
Wright State University, Dayton, Ohio

May 1999

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized data bases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***

- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at (301) 621-0134

- Telephone the NASA Access Help Desk at (301) 621-0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076

PREFACE

This report presents major findings of the research program entitled, "Reliability Based Structural Optimization," sponsored by NASA Lewis Research Center, Grant No. NAG 3 - 1489. Mr. Dale Hopkins was the NASA Technical Officer.

This report is intended for the demonstration of function approximation concepts and their applicability in reliability analysis and design. Particularly, approximations in the calculation of the safety index, failure probability and structural optimization (modification of design variables) are developed. With this scope in mind, extensive details on probability theory are avoided. Definitions relevant to the stated objectives have been taken from standard text books.

The idea of function approximations is to minimize the repetitive use of computationally intensive calculations by replacing them with simpler closed-form equations, which could be nonlinear. Typically, the approximations provide good accuracy around the points where they are constructed, and they need to be periodically updated to extend their utility.

There are approximations in calculating the failure probability of a limit state function. The first one, which is most commonly discussed, is how the limit state is approximated at the design point. Most of the time this could be a first-order Taylor series expansion, also known as the First Order Reliability Method (FORM), or a second-order Taylor series expansion (paraboloid), also known as the Second Order Reliability Method (SORM). From the computational procedure point of view, this step comes after the design point identification; however, the order of approximation for the probability of failure calculation is discussed first, and it is denoted by either FORM or SORM.

The other approximation of interest is how the design point, or the most probable failure point (MPP), is identified. For iteratively finding this point, again the limit

σ	Vector of Standard Deviations
σ_{x_i}	Standard Deviation of Random Variable x_i
$\sigma_{x_i'}$	Standard Deviation of Random Variable x_i'
σ_g	Standard Deviation of Function g
σ_R	Standard Deviation of Random Variable R
σ_S	Standard Deviation of Random Variable S
ρ_{ij}	Correlation Coefficient of x_i and x_j
s	Coefficient of Coordinate Shifting
S	Stress Random Variable
$\hat{S}$	Stress Random Variable
t_0	Scale Parameter of Type-I Extreme Value Distribution
$cos\theta_{x_i}$	Direction Cosine of the Safety Index Vector
U	Vector of Standard Normal Random Variables
$Var(Z)$	Variance of Function Z
Ω	Failure Region
X	Vector of Original (basic) Random Variables or Design Variables of Optimization
x_i	ith Original Random Variable
x_i'	ith Equivalent Normal Random Variable
$\hat{x}_i$	Sample Value of x_i
Y	Vector of Intervening Variables
Z	Linear Function of Random Variables X

state is approximated. The accuracy and efficiency of the approximations make the search process quite practical for analysis intensive approaches such as the finite element methods; therefore, the crux of this research is to develop excellent approximations for MPP identification and also different approximations including the higher-order reliability methods (HORM) for representing the failure surface.

This report is divided into several parts to emphasize different segments of the structural reliability analysis and design. Broadly, it consists of mathematical foundations, methods and applications. Chapter 1 discusses the fundamental definitions of the probability theory, which are mostly available in standard text books. Probability density function descriptions relevant to this work are addressed. In Chapter 2, the concept and utility of function approximation are discussed for a general application in engineering analysis. Various forms of function representations and the latest developments in nonlinear adaptive approximations are presented with comparison studies.

Research work accomplished in reliability analysis is presented in Chapter 3. First, the definition of safety index and most probable point of failure are introduced. Efficient ways of computing the safety index with a fewer number of iterations is emphasized. In Chapter 4, the probability of failure prediction is presented using first-order, second-order and higher-order methods. System reliability methods are discussed in Chapter 5. Chapter 6 presents optimization techniques for the modification and redistribution of structural sizes for improving the structural reliability.

This report also contains several appendices on probability parameters.

NASA/CR—1999-209154

Structural Reliability Analysis and Optimization: Use of Approximations

Ramana V. Grandhi and Liping Wang
Wright State University, Dayton, Ohio

Prepared under Grant NAG3–1489

National Aeronautics and
Space Administration

Glenn Research Center

May 1999

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076
Price Code: A15

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100
Price Code: A15

CONTENTS

VII. APPENDIX

List of Tables

List of Figures

NOMENCLATURE

α_i	Sensitivity Factor
α_s	Step Length
b	Vector of Reliability-based Optimization Design Variables
β	Reliability Index, Safety Index
β_c	Cornell Reliability Index, Safety Index
c_i	ith Constant of Linear Function Z
c_x	Coefficient of Variation
d_{max}	Allowable Maximum Displacement
D	Direction Vector
δ	Location Parameter of Type-I Extreme Value Distribution
$E(Z)$	Expected Value of Function Z
$f(b)$	Objective Function of Optimization
$FOSM$	First-Order Second-Moment Method
$f_{RS}(.)$	Joint Probability Density Function of Random Variables R and S
$G_j(X)$	jth Constraint Function of Deterministic Optimization Problem
$g(X)$	Original (X-space) Limit State Function
$\tilde{g}(X)$	Approximate Original (X-space) Limit State Function
$g(U)$	U-space Limit State Function
$\nabla g(U)$	Vector of Gradients of $g(U)$
$\tilde{g}(U)$	Approximate U-space Limit State Function
g'	First-order Gradient Vector of g Function
g''	Second-order Gradients of g Function
J	Number of Constraints
μ	Vector of Mean Values

μ_{x_i}	Mean Value of Random Variable x_i
$\mu_{x_i'}$	Mean Value of Random Variable x_i'
μ_g	Mean Value of Function g
μ_R	Mean Value of Random Variable R
μ_S	Mean Value of Random Variable S
m	Number of Failure Trials in Monte Carlo Simulation
M_i	ith Safety Index Margin
MPP	Most Probable Failure Point
n	Number of Random Variables
$\hat{n}$	Statistical Sample Size
N	Number of Design Variables
$\hat{N}$	Total Number of Trials in Monte Carlo Simulation
P	Structural Reliability (Probability)
P_f	Probability of Failure (system)
P_i^f	Element Failure Probability
P_{ij}^f	Joint Failure Probability
$\Phi(.)$	Standard Normal Cumulative Distribution Function
$\Phi_2(.)$	Two Dimensional Standardized Normal Distribution Function
$\phi(.)$	Standard Normal Density Function
$\phi_2(.)$	Two Dimensional Probability Density Function
p_i	Nonlinear Index of ith Variable
r	Nonlinear Index for All Variables
R	Strength Random Variable
ρ	Correlation Coefficient
$\vec{\rho}$	Correlation Matrix
$\hat{R}$	Strength Random Variable of Standard Normal Distribution

CHAPTER 1. PROBABILITY THEORY PRELIMINARIES

During the last ten years, there has been an increasing trend for analyzing structures using probabilistic information of loads, geometry, material properties, and boundary conditions. As the structures become more complex (e.g., space shuttle main engine parts, space structures, advanced tactical fighters, etc.) and the performance requirements become more ambitious, the need for analysis of uncertainties and computation of probabilities grows.

The reliability of a structure is its ability to fulfill its design purpose for a specified reference period. Most structures have a number of possible failure modes. In calculating the structural reliability, the influence of multiple disciplines has to be taken into account. Many structural problems are modeled and simulated using the finite element methods (FEM) for obtaining a detailed structural response. FEM is a computationally intensive numerical procedure with a large number of degrees of freedom. With an increase in the complexity of the structural model and the multidisciplinary nature of analyses, the number of failure modes and their computation increase very significantly. For accurate and efficient calculation of structural reliability, new and innovative methods have to be employed to make this performance measure practical in a wide variety of structural applications. The structural reliability can be used as a comparative measure in choosing among competitive designs.

In this chapter, basic definitions of probability theory, density function distributions and design issues are discussed.

1.1 Reliability and Its Importance

Reliability is the probability of a system performing its function over a specified period of time and under specified service conditions. Structural response depends on many factors such as loads, boundary conditions, stiffness and mass properties. The response is considered satisfactory when the design requirements imposed on the structural behavior are met. Each of

these requirements is termed as "limit state" or "constraint". The study of *structural reliability* concerns the calculation and prediction of the probability of limit state violation at any stage during its life. The probability of occurrence of an event such as limit state violation is a numerical measure of the chance of its occurring. The next goal in this calculation is to improve the structural reliability to minimize the risk and failure with the available and allowed design alternatives.

1.1.1 Factor of Safety and Reliability

Factor of safety is used to maintain a proper degree of safety in structural design. Generally, the factor of safety is understood to be the ratio of the expected strength to the expected load. In practice, both the strength and load are variables, the values of which are scattered about their respective mean values. When the scatter in variables is considered, the factor of safety could potentially be less than unity, and the traditional factor of safety based design would fail.

1.2 Probability Theory Introduction

An *experiment* denotes the act of performing something the outcome of which is subject to uncertainty and not known exactly. For example, tossing a coin, rolling a die, etc. The *sample space* is the set of all the possible outcomes of the experiment, denoted as S. The sample space can be discrete or continuous. An *event* is the outcome of a single experiment. For example, realizing a head on tossing a coin, getting an even number (2 or 4 or 6) on rolling a die. The union of two events A and B is written as $A \cup B$ and is the set of outcomes that belong to A or B or both. The intersection of the two events A and B is written as $A \cap B$ and is the set of outcomes that belongs to both A and B. A null event (empty set) has no outcomes. If the occurrence of one *event* precludes the occurrence of other events in a given experiment, the events are called *mutually exclusive*. The complement of event A is written as $\bar{A}$ and is the outcomes of S which do not belong to A.

1.2.1 Definition of Probability

The probability of occurrence of an event E is defined as the ratio of the number of occurrences of E to the total number of trials.

$$P(E) = lim(\frac{n}{N}) \tag{1.1}$$

where n is the number of trials in which the event E has occurred, and N is the total number of trials. Also,

$$0 \leq P(E) \leq 1 \tag{1.2}$$

$$P(E) = 0 \quad E \text{ is impossible} \tag{1.3}$$

$$P(E) = 1 \quad E \text{ is certain} \tag{1.4}$$

If A and B are mutually exclusive,

$$P(A \cup B) = P(A) + P(B) \tag{1.5}$$

In general, if A and B are any two events, then

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) \tag{1.6}$$

$$P(A) = 1 - P(\bar{A}) \tag{1.7}$$

1.2.1.1 Conditional Probability

Conditional probability of events A and B is the probability of B given that A has occurred. It is written as $P(B|A)$.

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \tag{1.8}$$

From the definition of conditional probability,

$$P(A \cap B) = P(B|A)P(A) \tag{1.9}$$

$$P(B \cap A) = P(A|B)P(B) \tag{1.10}$$

Events A and B are said to be independent if and only if

$$P(A \cap B) = P(A)P(B) \tag{1.11}$$

Example 1.1

A pair of ordinary dice are thrown. What is the probability of the sum of spots on the upward-landing faces being 7 (event A), given that this sum is odd (event B)?
The sample space is composed of 36 outcomes:

$$\begin{aligned}\Omega \;=\; \{&(1,1),(1,2),(1,3),(1,4),(1,5),(1,6),\\ &(2,1),(2,2),(2,3),(2,4),(2,5),(2,6),\\ &(3,1),(3,2),(3,3),(3,4),(3,5),(3,6),\\ &(4,1),(4,2),(4,3),(4,4),(4,5),(4,6),\\ &(5,1),(5,2),(5,3),(5,4),(5,5),(5,6),\\ &(6,1),(6,2),(6,3),(6,4),(6,5),(6,6)\}\end{aligned}$$

The number of outcomes favorable to A is 6, so the unconditional probability is

$$P(A) = \frac{6}{36} = \frac{1}{6}$$

If B has occurred, then one of 18 events occurred (a "new" sample space with 18 points), and the conditional probability is

$$P(A|B) = \frac{6}{18} = \frac{1}{3}$$

The probability of event B is

$$P(B) = \frac{18}{36} = \frac{1}{2}$$

and $P(A|B)$ can also be obtained from Eq.(1.8):

$$P(A|B) = \frac{P(B \cap A)}{P(B)} = \frac{\frac{6}{36}}{\frac{1}{2}} = \frac{1}{3}$$

1.2.2 Random Variable

A random variable X, takes on various values x within the range $-\infty < x < \infty$. A random variable is denoted by a capital letter, and its particular value is represented by a lower case letter. Random variables are of two types; (i) discrete and (ii) continuous. If the random variable is allowed to take only discrete values $x_1, x_2, \ldots, x_n$, it is called a discrete random variable. On the other hand, if the random variable is permitted to take any real value in a specified range, it is called a continuous random variable. In this report, we concentrate on continuous variables. For example, the yield strength R of steel is a random variable. When R is measured in a tensile test, different values are observed for each identically prepared specimen

$$0 < R < \infty \tag{1.12}$$

1.2.3 Probability Density and Cumulative Distribution functions

The function that describes the distribution of a random variable over the sample space of the continuous random variable, X, is called the probability density function (pdf) and is designated as $f_X(x)$. The cumulative distribution function (cdf) $F_X(x)$ is an alternate way to describe the probability distribution for both discrete and continuous random variables. cdf is defined for all values of random variable X from $-\infty$ to $+\infty$ and is equal to the probability that X is less than or equal to a specific value x.

For a continuous random variable, $F_X(x)$ is calculated by integrating the pdf for all values of X less than or equal to x:

$$F_X(x) = \int_{-\infty}^{x} f_X(s)ds \tag{1.13}$$

If the random variable X is continuous and if the first derivative of the distribution function exists, the probability density function $f_X(x)$ is given by the first derivative of $F_X(x)$:

$$f_X(x) = \frac{dF_X(x)}{dx} \tag{1.14}$$

The cdf is a nondecreasing function of x (its slope is always greater than or equal to zero) with lower and upper limits of 0 and 1, respectively. cdf is also referred to in the report as a distribution function. A typical probability density function and the corresponding distribution function are shown in Fig. 1.1.

In general, there are n random variables. The outcome is an n dimensional random vector. The probability is calculated as

$$P[a < X < b, c < Y < d] = \int_c^d \int_a^b f_{XY}(x,y)dxdy \tag{1.15}$$

Properties:

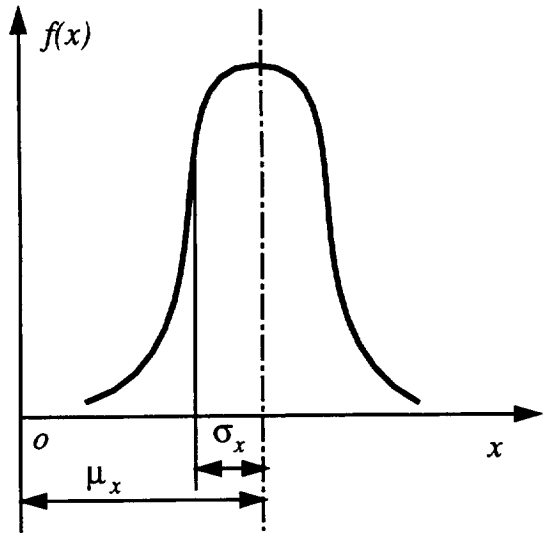
$$f_{XY}(x,y) \geq 0 \tag{1.16a}$$

$$\int_s \int f_{XY}(x,y)dxdy = 1 \tag{1.16b}$$

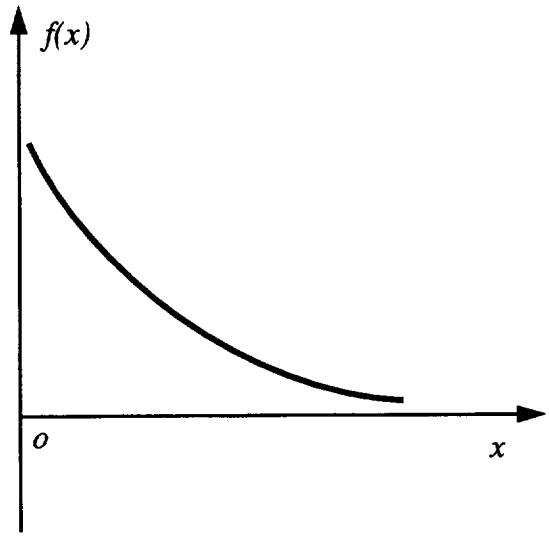
1.2.3.1 Joint Density and Distribution Functions

For independent random variables, the joint density function is given by the product of individual or marginal density functions as

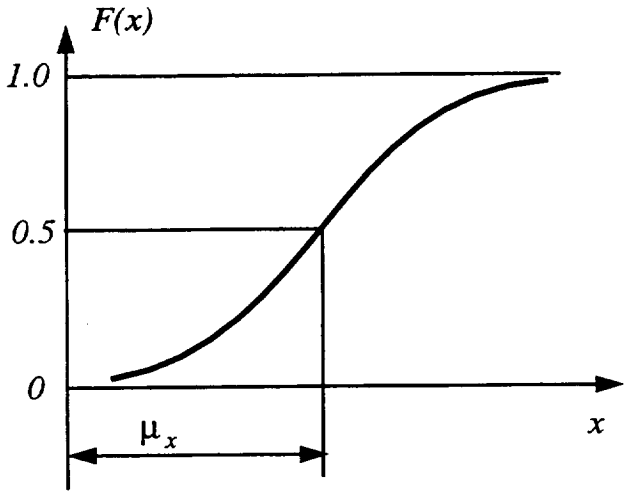
$$f_{X_1,X_2,\ldots,X_3} = f_{X_1}(x_1)\ldots f_{X_n}(x_n) \tag{1.17}$$



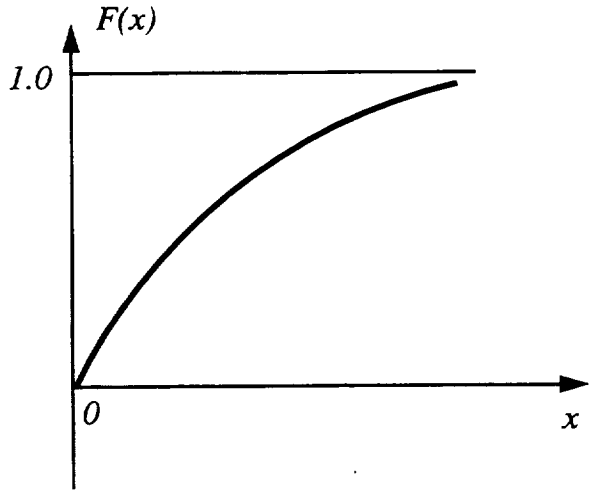
(a) Normal density function



(a) Exponential density function



(b) Normal distribution function



(b) Exponential distribution function

Fig. 1.1 Probability Density and Distribution Functions

The joint distribution function is given as

$$F_{X_1,X_2,\ldots,X_n}(x_1, x_2, \ldots x_n) = F_{X_1}(x_1).F_{X_2}(x_2)\ldots F_{X_n}(x_n) \tag{1.18}$$

In general, to compute a probability associated with an event involving a multidimensional vector whose joint pdf is $f_X(X)$,

$$P(X \in \Omega) = \int_\Omega f_X(X)dX \tag{1.19}$$

This is the probability that the random vector X will lie in region Ω.

1.2.3.2 Marginal Distribution of X

For example, the pdf of X without regard to Y is

$$f_X(x) = \int_{-\infty}^{\infty} f_{XY}(x, y)dy \tag{1.20}$$

1.2.3.3 Conditional Distribution of X given Y

The pdf of X for a specified Y is

$$f_{X|Y}(x|y) = \frac{f_{XY}(x, y)}{f_Y(y)} \qquad f_Y > 0 \tag{1.21}$$

1.2.3.4 Independent Random Variables

If X and Y are independent, then,

$$f_{X|Y}(x, y) = f_X(x) \tag{1.22}$$

The conditional pdf becomes the marginal pdf, and the joint pdf becomes the product of the marginals:

$$f_{XY}(x, y) = f_X(x)f_Y(y) \tag{1.23}$$

In general, the joint pdf is equal to the product of the marginals when all variables are mutually independent:

$$f_X(X) = \prod_{i=1}^{n} f_{X_i}(x_i) \tag{1.24}$$

1.2.4 Mean, Mode and Median

The probability density or distribution function of a random variable contains complete information about the variable. However, in many cases, the gross properties of the variable are used. The most commonly used are the mean and standard deviation. A measure of central tendency is captured by the mean value μ of the probability distribution. The variation from the mean is captured, to first order, by the variance σ^2 and its by products, the standard deviation σ and coefficient of variation C_X.

Mean(first moment):

The mean value, also termed as the expected value or average, is used to describe the central tendency of a random variable. This is a "weighted average" of all the values that a random variable may take. If $f_X(x)$ is the probability density function of X, the mean is given by

$$\mu = E(X) = \int_{-\infty}^{\infty} x f_X(x) dx \tag{1.25}$$

μ is the distance to the centroid of the pdf. It is called the "first moment" since it is first moment of area of the pdf. The mean is analogous to the centroidal distance of a cross-section.

According to the definition of a random variable, any function of a random variable is itself a random variable. Therefore, if $g(x)$ is an arbitrary function of x, the expected value of $g(x)$ is defined as

$$E[g(x)] = \int_{-\infty}^{\infty} g(X) f_X(x) dx \tag{1.26}$$

Mode:

Mode is the value of X corresponding to the peak value of the probability density function.

Median:

Median is the value of X at which the cumulative distribution function has a value of 0.5.

Higher-Order Moments

Define the n^{th} order moment by letting $Y = X^n$:

$$E(X^n) = \int_{-\infty}^{\infty} x^n f_X(x) dx \tag{1.27}$$

For $n = 2$, the mean square value of X is

$$E(X^2) = \int_{-\infty}^{\infty} x^2 f_X(x) dx \tag{1.28}$$

Properties of Expected Values

$E(cX) = cE(X)$

where c is a constant.

Given $Y = X_1 + X_2 + ...X_n$, the expected value of Y is a linear combination of individual values:

$E(Y) = E(X_1) + E(X_2) + ...E(X_n)$

Only if X and Y are independent,

$E(XY) = E(X)E(Y)$

1.2.5 Standard Deviation and Skewness Coefficient

The expected value or mean value is a measure of the central tendency, which indicates the location of the distribution on the coordinate axis representing the random variable. A measure of the variability of the random variable is usually given by a quantity known as the standard deviation. Another quantity, which not only gives a measure of the variability, but also a measure of the symmetry of the density function, is called the *skewness coefficient.*

1.2.5.1 Standard Deviation

The variance of a random variable X is as a measure of the degree of randomness about the mean

$$\sigma_X^2 = V(X) = E[(X - \mu_X)^2] \tag{1.29}$$

Geometrically, it represents the moment of inertia of the pdf about the mean value. The variance of a random variable is analogous to the moment of inertia of a weight about its centroid. The variance or standard deviation is a measure of the variability of a random variable or the breadth of the density function. The standard deviation is defined as

$$\sigma_X = +\sqrt{V(X)} \tag{1.30}$$

The standard deviation is often preferred over the variance as a measure of dispersion because it has the same units as X and μ.

Properties of Variance

$V(X)$ can also be written as $E(X^2) - \mu_X^2$. If c is a constant, $V(c + X) = V(X)$, $V(X_1 + X_2 + ... + X_n) = V(X_1) + V(X_2) + ... + V(X_n)$, only if X_i are mutually independent. The coefficient of variation is a measure of dispersion in nondimensional form and is defined as

$$coefficient \quad of \quad variation \quad of \quad X = \frac{standard \quad deviation}{mean} = \frac{\sigma_X}{\mu_X} \tag{1.31}$$

1.2.5.2 Skewness Coefficient

The expected value of the cube of the deviation of the random variable from its mean value (also known as the third moment of the distribution about the mean) is taken as a measure of the skewness or lack of symmetry of the distribution:

$$E[(X - \mu_X)^3] = \int_{-\infty}^{\infty} (X - \mu_X)^3 f_X(x) dx \tag{1.32}$$

The value of $E[(X-\mu_X)^3]$ can be positive or negative. The skewness coefficient is defined as

$$skewness \quad coefficient = \frac{E[(X-\mu_X)^3]}{\sigma_X^3} \tag{1.33}$$

1.2.5.3 Covariance

The covariance of two random variables X and Y is defined as

$$Cov(X,Y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\mu_X)(y-\mu_Y).f_{X,Y}(x,y)dxdy = \sigma_{XY} \tag{1.34}$$

By expanding the product, it can be rewritten as

$$\sigma_{XY} = E(XY) - \mu_X\mu_Y \tag{1.35}$$

where

$$E(XY) = \int_{-\infty}^{\infty} xyf_{XY}(x,y)dxdy \tag{1.36}$$

If X and Y are independent variables, then $\sigma_{XY} = 0$. The converse is not generally true. If $\sigma_{XY} = 0$, X and Y are said to be uncorrelated.

The correlation coefficient $\rho_{X,Y}$ for the random variables is defined as

$$\rho_{X,Y} = \frac{Cov(X,Y)}{\sigma_X.\sigma_y} \tag{1.37}$$

and its value lies between -1 and 1. The correlation coefficient is often used to characterize the relationship between two variables. The physical meaning of the correlation coefficient is that its value is nearly unit if the two random variables are linearly related, but is nearly zero if they are not.

1.3 Probability Distributions

There are several types of probability distributions for describing random variables. The selection of a particular type of probability distribution depends on (i) the nature of the problem,

(ii) the underlying assumptions associated with the distribution, (iii) the shape of the curve between $f(x)$ or $F(x)$ and x obtained after plotting the available data, and (iv) the convenience and simplicity afforded by the distribution in subsequent computations.

The properties of some of the more commonly used distributions are presented in the following sections.

1.3.1 Normal Distribution

The density function of a normally distributed random variable X (also known as Gaussian distribution) is given by

$$f_X(x) = \frac{1}{\sqrt{2\pi}\sigma_X} exp[-\frac{1}{2}(\frac{x-\mu_x}{\sigma_X})^2] \tag{1.38}$$

where X is identified as $N(\mu_X, \sigma_X)$. The parameters of the distribution μ_X and σ_X denote, respectively, the mean value and standard deviation of the variable X. The density function and the corresponding distribution function are shown in Fig. 1.1. The normal distribution has the following properties:

(i) Any linear function of normally distributed random variables is also normally distributed.

Let Z be the sum of normally distributed random variables

$$Z = a_0 + a_1X_1 + a_2X_2 + ... + a_nX_n \tag{1.39}$$

where a_i's are constants. Then Z will be normal, where

$$\mu_z = a_0 + \sum_{i=1}^{n} a_i\mu_i \qquad \sigma_z = \sqrt{\sum_{i=1}^{n}(a_i\sigma_i)^2} \tag{1.40}$$

(ii) The nonlinear function of normally distributed random variables can be normal, Weibull, gamma, lognormal, etc. or example, the function $y = \sqrt{X_1^2 + X_2^2}$ of two independent and standard normally distributed random variables X_1 and X_2 with $N(0, \sigma^2)$ is a Rayleigh distribution function as shown in Fig. 1.2. Its density and distribution functions are

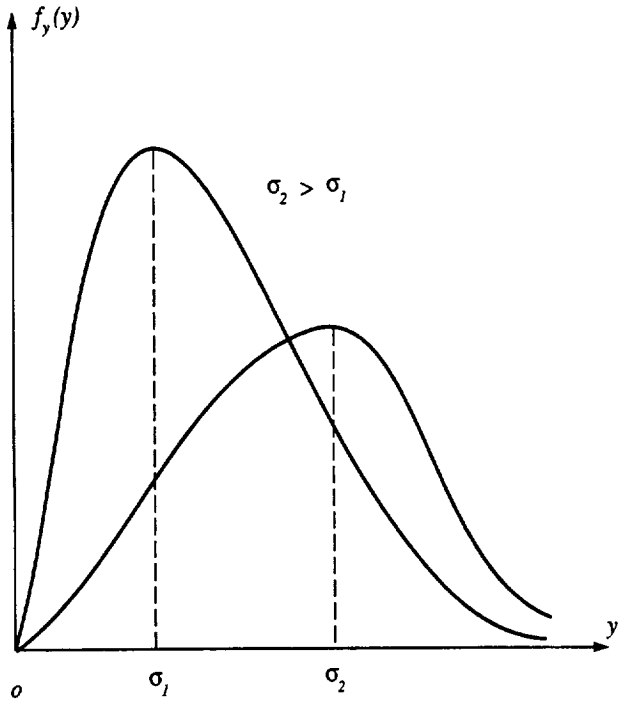


Fig. 1.2 Rayleigh Distribution with $\sigma=\sigma_1$ & σ_2

computed as

$$f_y(y) = \begin{cases} \frac{y}{\sigma} e^{-\frac{y^2}{2\sigma^2}}, & y \geq 0 \\ 0 & otherwise \end{cases} \tag{1.41a}$$

$$F_y(y) = \begin{cases} 1 - e^{-\frac{y^2}{2\sigma^2}}, & y \geq 0 \\ 0 & otherwise \end{cases} \tag{1.41b}$$

1.3.2 Standard Normal Distribution

A Gaussian distribution with parameters $\mu = 0$ and $\sigma = 1$ is called the standard normal distribution and is identified as $N(0,1)$. The density function of a standard normal variate (variable) Z is given by

$$f_Z(z) = \frac{1}{\sqrt{2\pi}} exp[-\frac{z^2}{2}] \qquad -\infty < z < \infty \tag{1.42}$$

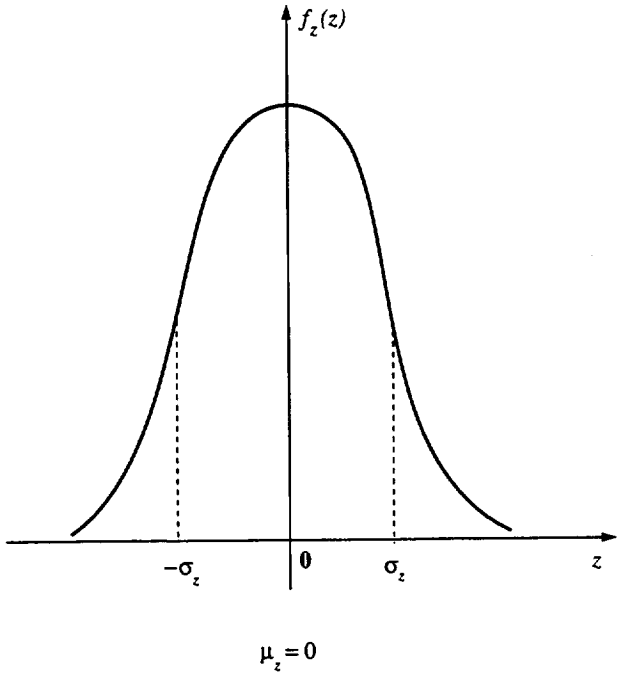


Fig. 1.3 Standard Normal Distribution

and it is symmetric about its mean $\mu = 0$ as shown in Fig. 1.3. The distribution function of the standard normal variate Z is commonly denoted as $\Phi(z)$ and is given by

$$\Phi(z) = F_Z(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} exp[-\frac{z^2}{2}]dz \tag{1.43}$$

If $\Phi(z_p) = p$ is given, the standard normal variate z_p corresponding to the cumulative probability (p) is denoted as

$$z_p = \Phi^{-1}(p) \tag{1.44}$$

The values of the distribution function $\Phi(z)$ of a standard normal variate are given as tables of normal distribution (Ref. [4]). Usually, the possibilities are given in tables only for positive values of z and for negative values

$$\Phi(-z) = 1 - \Phi(z) \tag{1.45}$$

due to the symmetry of the density function about zero. Similarly, we can find that

$$z_p = \Phi^{-1}(p) = -\Phi^{-1}(1-p) \tag{1.46}$$

Once the standard normal table of $\Phi(z)$ is available, the probabilities of any other normal distribution can be determined using the following procedure. For a nonstandard X with $N(\mu, \sigma)$

$$P(\ell \leq X \leq u) = \frac{1}{\sqrt{2\pi}\sigma} \int_{\ell}^{u} exp[-\frac{1}{2}(\frac{x-\mu_x}{\sigma_X})^2].dx \tag{1.47}$$

This represents the area under the density function between ℓ and u. By defining a new variable (standard normal variate) z as

$$z = \frac{x-\mu}{\sigma}$$

then

$$dz = \frac{dx}{\sigma} \tag{1.48}$$

and the probability becomes

$$P(\ell \leq X \leq u) = \frac{1}{\sqrt{2\pi}} \int_{\frac{(\ell-\mu)}{\sigma}}^{\frac{(u-\mu)}{\sigma}} exp[-\frac{z^2}{2}].dz \tag{1.49}$$

which can be recognized as the area under the standard normal density function between $(\frac{u-\mu}{\sigma})$ and $(\frac{\ell-\mu}{\sigma})$. Thus, the required probability can be found as

$$P(\ell \leq X \leq u) = \Phi(\frac{u-\mu}{\sigma}) - \Phi(\frac{\ell-\mu}{\sigma}) \tag{1.50}$$

1.3.3 Lognormal Distribution

A random variable X is said to follow lognormal distribution (Fig. 1.4) if $Y = lnX$ follows normal distribution. Thus,

$$f(y) = \frac{1}{\sqrt{2\pi}\sigma_Y} exp[-\frac{1}{2}(\frac{y-\mu_Y}{\sigma_Y})^2] \quad -\infty < y < \infty \tag{1.51}$$

Since $Y = lnX$ the above equation can be rewritten in terms of X as

$$f(x) = \frac{1}{\sqrt{2\pi}x\sigma_Y} exp[-\frac{1}{2}(\frac{ln\ x-\mu_Y}{\sigma_Y})^2] \quad x \geq 0 \tag{1.52}$$

where

$$\sigma_Y^2 = ln[(\frac{\sigma_x}{\mu_X})^2 + 1] \tag{1.53}$$

and

$$\mu_Y = ln\ \mu_x - \frac{1}{2}\sigma_Y^2 \tag{1.54}$$

1.3.4 Weibull Distribution

The probability density function (Fig. 1.5) is

$$f_X(x) = \frac{\alpha x^{\alpha-1}}{\beta^\alpha} exp[-(\frac{x}{\beta})^\alpha] \quad x > 0, \alpha > 0, \beta > 0 \tag{1.55}$$

and the distribution function cdf is

$$F_X(x) = 1 - exp[-(\frac{x}{\beta})^\alpha] \tag{1.56}$$

This is a two parameter family, α and β. The moments in terms of the parameters are

$$E(X^n) = \beta^n \Gamma(\frac{n}{\alpha} + 1) \tag{1.57}$$

where $\Gamma(.)$ is the gamma function. The mean and coefficient of variation are

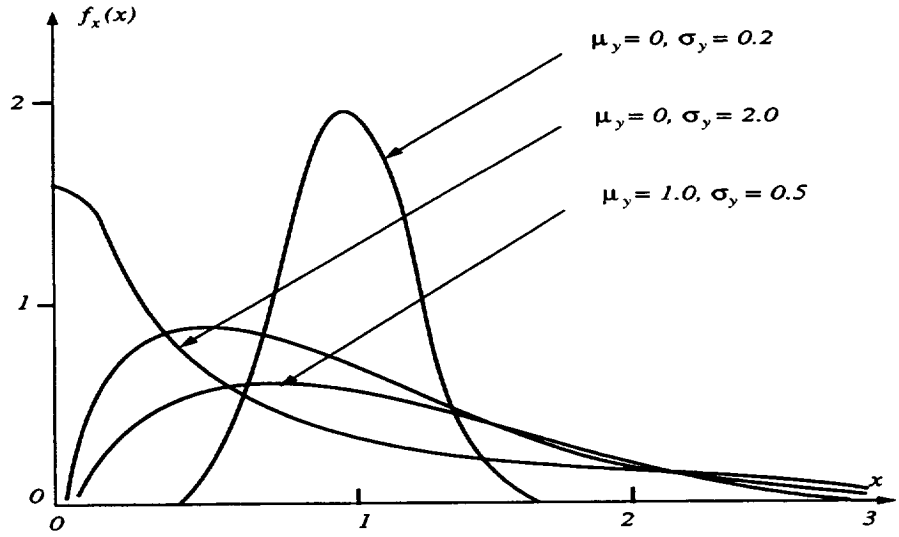


Fig. 1.4 Log–normal Density Function

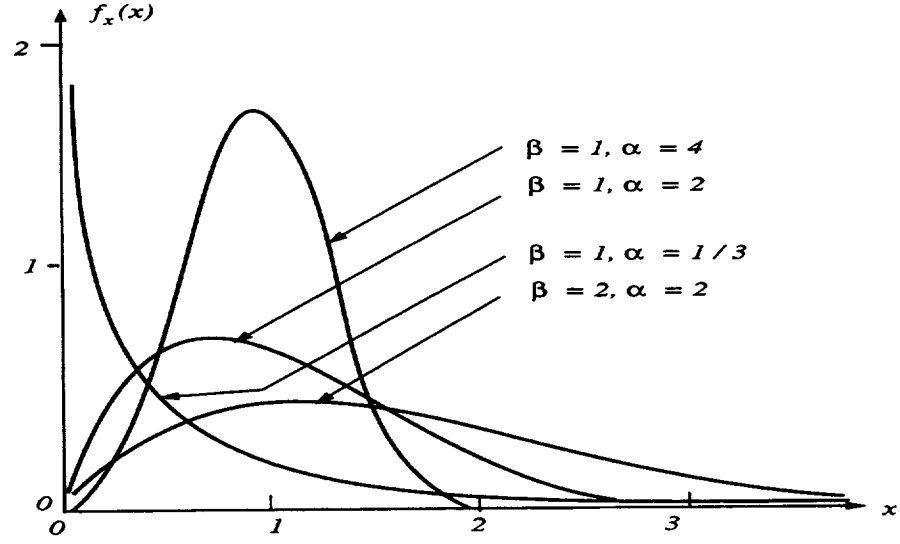


Fig. 1.5 Weibull Density Function

$$\mu_X = \beta\Gamma(\frac{1}{\alpha}+1) \tag{1.58}$$

$$C_X = [\frac{\Gamma(\frac{2}{\alpha}+1)}{\Gamma^2(\frac{1}{\alpha}+1)} - 1]^{0.5} \tag{1.59}$$

The mean and standard deviation are complicated functions of the parameters. However, the approximation, $\alpha = C_X^{-1.08}$ is a very good one (over the range of interest to engineers). The following parameters are recommended in Ref. [6].

$$\alpha = C_X^{-1.08} \qquad \beta = \frac{\mu_X}{\Gamma(\frac{1}{\alpha}+1)} \tag{1.60}$$

1.3.5 Exponential Distribution

This is a special case of Weibull distribution for $\alpha = 1$. The pdf is

$$\begin{aligned} f_X(x) &= \lambda exp[-\lambda x] \quad x > 0 \\ &= o \quad otherwise \end{aligned} \tag{1.61}$$

The cdf is

$$F_X(x) = 1 - exp[-\lambda x] \tag{1.62}$$

The moments in terms of the parameter λ are

$$\mu_x = \frac{1}{\lambda} \qquad \sigma_x = \frac{1}{\lambda} \tag{1.63}$$

The exponential is commonly used in reliability analysis. If time to failure T of a unit has an exponential distribution, λ represents failure rate (occurrences/time). This is commonly used in electrical engineering and is occasionally used in design, e.g., long-term distribution of fatigue stresses.

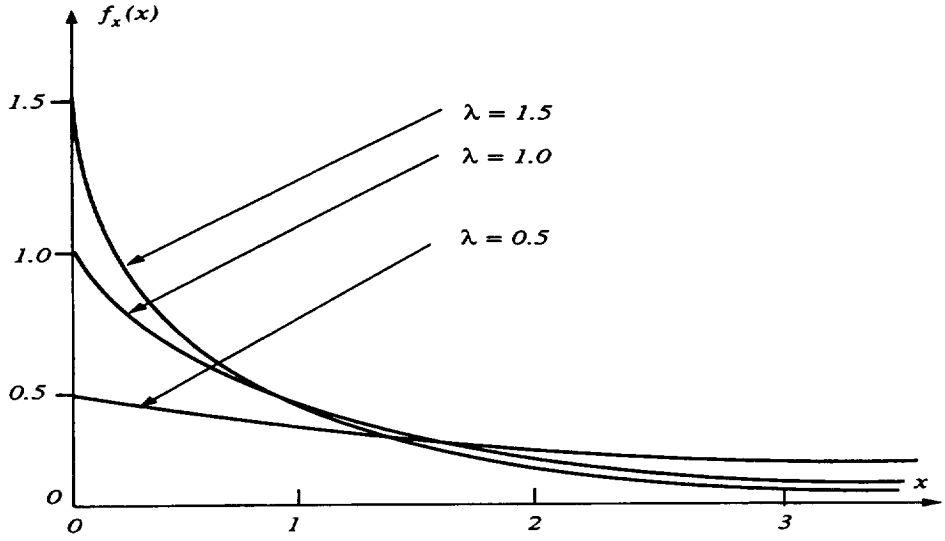


Fig. 1.6 Exponential Density Function

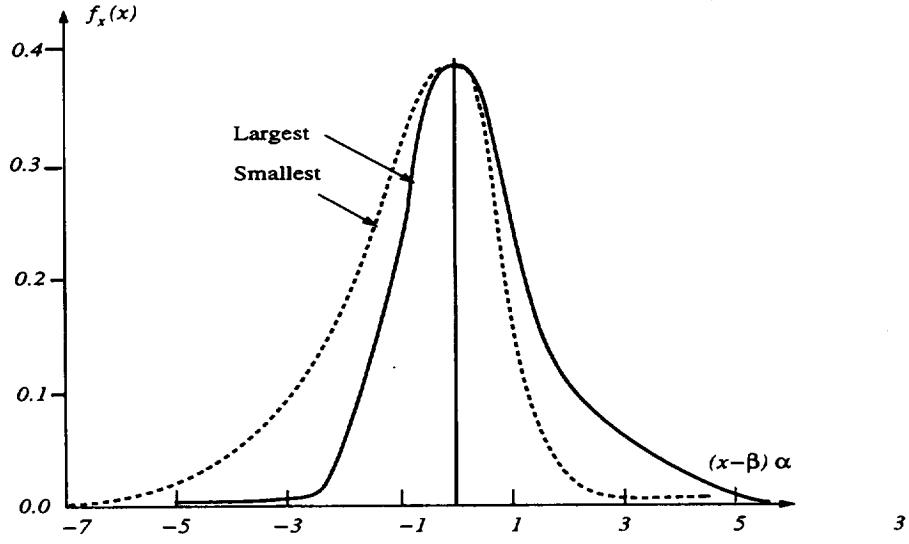


Fig. 1.7 Extreme Value Density Function (Type I)

1.3.6 Extreme Value Distribution-Type I Distribution of Maxima

The cdf is

$$F_X(x) = exp[-[exp\{-\alpha(x-\beta)\}]] \tag{1.64}$$

The parameters are α and β. The mean and standard deviation in terms of α and β are

$$\mu_x = \beta + \frac{0.577}{\alpha}$$
$$\alpha_x = \frac{1.283}{\alpha} \tag{1.65}$$

Typical examples of extreme value distribution are to describe loading such as:

(i) the random variable describing the peak gust velocity experienced by an aircraft in very long hours of operation, and

(ii) The random variable denoting the maximum water level in a year at a point in a stream.

1.4 Choice of a Statistical Model

1.4.1 Normal Distribution

For small coefficients of variation this can be used. Examples are modulus of elasticity, Poisson's ratio, material properties, *etc.*

1.4.2 Lognormal Distribution

This can be used for most variables and plays an important role in probabilistic design. Examples are cycles of failure in fatigue, material strengths, loading variables, *etc.*

1.4.3 Weibull Distribution

This is a very popular distribution, but it is probably over used. Examples are fatigue, material strength, time to failure in reliability analysis, and long-term distribution of stress

ranges in fatigue.

1.4.4 Extreme Value Distribution

Type I of maxima is used almost exclusively for extreme environmental (load) variables.

1.5 Normal Variables – Linear Response Function

For a response function Z of the form

$$Z = a_0 + \sum_{i=1}^{n} a_i X_i \tag{1.66}$$

where a_i are constants and the random variables X_i have normal distribution with a mean of μ_i and standard deviation of σ_i, Z is also normal (for any n). The mean and standard deviation of Z are

$$\mu_Z = a_0 + \sum_{i=1}^{n} a_i \mu_i \tag{1.67}$$

$$\sigma_Z = \sqrt{\sum_{i=1}^{n} a_i^2 \sigma_i^2} \tag{1.68}$$

1.6 Lognormal Variables – Multiplicative Response Function

For a response function Z of the form

$$Z = a_0 \prod_{i=1}^{n} X_i^{a_i} \tag{1.69}$$

where a_i are constants and the random variables X_i have lognormal distribution with a median of $\tilde{X}_i$ and coefficient of variation C_{X_i}. It is more convenient to use the median and coefficient of variation as the basic parameters for lognormal variates.

The resulting response function Z is also lognormally distributed with median,

$$\tilde{Z} = a_0 \prod_{i=1}^{n} \tilde{X}_i^{a_i} \tag{1.70}$$

and the coefficient of variation

$$C_U = \sqrt{\prod_{i=1}^{n}(1 + C_{X_i}^2)^{a_i^2} - 1} \tag{1.71}$$

1.7 Uncertainties in the Design Process

1.7.1 Strength or Resistance (R) Uncertainties

(i) The exact strength properties of the material are unknown.

(ii) The size effects are not accurately known.

(iii) The effects of machining and processing operations on the strength are not known.

(iv) There is uncertainty of the effect of the assembly operations on the strength of the system.

(v) The effect of time on the strength is not known.

1.7.2 Stress or Loading (S) Uncertainties

(i) The assumptions used in modeling and stress analysis contain errors. Discontinuities and stress concentrations are often ignored in the analysis.

(ii) The magnitude of the peak loads are not exactly known.

1.8 Probabilistic Design

Probabilistic and statistical methods are convenient tools to describe or model physical phenomena that are too complex to treat with the present level of scientific knowledge. Probabilistic design procedures promise to improve the product quality of engineering systems for the following reasons. Probabilistic design explicitly incorporates given statistical data into the design algorithms, whereas conventional design discards such data. Rational comparisons can be made between two or more competing designs for a proposed system. In the absence of other considerations, the engineer chooses the design having the lowest failure probability.

Probabilistic based information on mechanical performance can be used to develop rational policies towards pricing, warranties, spare parts requirements, etc.

1.9 Summary

In this chapter, basic definitions of probability theory and distribution functions were introduced. Further details and explanations can be found in the text books cited in this chapter. In this report, only the definitions and terminology relevant to the following chapters were presented.

*

References

[1] Madsen, H.O., Krenk, S., and Lind, N.C., Methods of Structural Safety, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1986.

[2] Christensen, P.T. and Murotsu, Y., Application of Structural Systems Reliability Theory, Springer-Verlag, Berlin, 1986.

[3] Melchers, R.E., Structural Reliability, Analysis and Prediction, Ellis Horwood, Chichester, England, 1987.

[4] Dai S. H. and Wang M. O., Reliability Analysis in Engineering Applications, Van Nostrand Reinhold, 1992.

[5] Rao, S.S., Reliability-based Design, McGraw-Hill, Inc., New York, 1992.

[6] Wirsching, P. H. and Sundararajan, C., Probabilistic Structural Mechanics Handbook: Theory and Industrial Applications, Chapman & Hall, New York, 1995.

[7] Ditlevsen, O., Madsen, H.O., Structural Reliability Methods, John Wiley & Sons, Chichester, England, 1996.

CHAPTER 2. FUNCTION APPROXIMATION TOOLS

Function approximations have been playing a major role in optimization of large scale structures during the last two decades. Since their inception in the 1970s by Schmit and Farshi [1], they have found an important place in the research, implementation and technology transfer of many structural optimization algorithms. For many structural optimization problems, the evaluation of the objective function and constraints requires the execution of costly finite element analyses for displacements, stresses or other structural responses. The optimization process may require evaluating the objective function and constraints hundreds or thousands of times. The cost of repeating the finite element analysis so many times is usually prohibitive. However, this computational cost problem can be addressed by the use of approximations during portions of the optimization process. First, an initial design is obtained by using an exact analysis, and the information needed in constructing the approximations is generated. The original optimization problem is changed into a sequential approximate optimization problem with approximate constraints. Then, the approximate problem is solved by an optimization algorithm. The objective function value is obtained at the optimum solution and compared with the initial value. If the convergency is not satisfied, the process is repeated until convergence. Since the approximation has replaced the expensive exact constraint calculations, significant computational savings can be realized, particularly for the large scale structural problems requiring time-consuming analyses.

Probabilistic structural analysis is an inherently computationally intensive procedure, and the problem is exacerbated by the convergence difficulties associated with highly nonlinear and large scale structural problems. To alleviate this problem, many approximate reliability methods have been developed within the past two decades. A commonly used approximate approach is to use uncertainties information represented by only the first two moments (mean and standard deviation) and change the original probability model (multidimensional

integration over an irregular region) into a safety index problem. This safety index problem is actually an optimization problem of finding a point on the structural response surface which has the shortest distance from the origin to the surface in the standard normal space. Therefore, the approximation approach in optimization can be applied to solve the safety index. Furthermore, the probability is computed based on the safety index information and the approximations of the limit state surface. According to different approximations of the surfaces, different probability methods are generated, such as the first-order reliability method (FORM), second-order reliability method (SORM) and higher-order reliability method (HORM).

Most of the approximations presented in the literature were based on function and gradient information at a single point and constructed by using the first-order Taylor series expansion about this point. This method is very popular because the function and its derivative values are always needed for the search direction calculation, so no additional computation is involved in developing an approximate function. There are several variations of first-order Taylor series approximations, most notably the linear, reciprocal and conservative approximations. These approximations work effectively for stress and displacement type problems; however, the truncation error of the first-order approximation might be large and could be inaccurate, even for design points closer to the expansion point. The accuracy of the first-order approximations may be increased in some disciplines by retaining higher-order terms in Taylor series expansion, such as the quadratic approximation. This requires the calculation of higher-order function derivatives that may not be available analytically.

Both the above first-order and higher-order approximations are formed by using the first-order and higher order Taylor series expansion, respectively, in terms of direct and reciprocal design variables. The intervening variables are fixed in these approximations. For example, the linear approximation can be considered as the first-order Taylor series expansion in terms of the intervening variables $y_i = x_i$. The reciprocal approximation is the first-order Taylor series expansion in terms of the intervening variables $y_i = \frac{1}{x_i}$. As we know, for the truss

structures with stress and displacement functions, using the reciprocal intervening variables can dramatically improve the approximation accuracy. However, the use of fixed intervening variables is difficult to adopt for different problems, and selection of the intervening variables is also quite difficult and requires tremendous experience and knowledge. Therefore, the use of adaptive intervening variables for different types of problems is necessary.

Furthermore, both the first-order and higher-order Taylor series approximations are based on a single point. As the structure is being resized, new approximations are constructed at new design points. In this approach, previous analyses' information is discarded and not used to improve the later approximations. Recently, more accurate approximations have been developed by using more than one data point, such as two points, three points or more. These multi-point approximations use current and previous information to construct approximations. Since more information about the function values and gradients at the known points are provided, the multi-point approximation is usually able to automatically adust its nonlinearities by itself. Therefore, the multi-point approximations are adaptive and provide better accuracy than the single point approximations. Also, no higher-order gradients are needed in constructing the approximations.

In this chapter, the use of approximations and advantages are discussed in Section 2.1. The availability of gradients is given in Section 2.2. The approximations constructed based on a single point are introduced in Section 2.3, which includes the three most commonly-used approximations (linear, reciprocal, and conservative approximations). The two-point approximations which were developed in Wang and Grandhi's earlier work are given in Section 2.4, which includes the two-point adaptive nonlinear approximation (TANA) and improved two-point adaptive nonlinear approximation (TANA2). The multi-point Hermite approximation is introduced in Section 2.5. The approximation comparisons of various approximations are discussed in Section 2.6. The relevant references for the approximations are listed in Section 2.7.

2.1 Use of Approximations and Advantages

As mentioned above, the function approximations are particularly useful when the computational cost of a single evaluation of the object functions, constraints, and their derivatives is very large compared to the computational cost associated with the optimization operations, such as the calculation of search directions. A typical situation is when a finite element model with thousands of a degrees of freedom is used to analyze a structural design that is defined in terms of a handful of design variables. It then pays to reduce the number of exact structural analyses required for the design process by applying optimization algorithms to a model of the structure based on approximations.

In general, the optimization problem is stated as

$$Minimize \qquad f(X) \tag{2.1a}$$

$$Subject \quad to: \qquad G_j(X) \leq 0, \qquad (j = 1, 2, ..., J) \tag{2.1b}$$

$$x_i^L \leq x_i \leq x_i^U \qquad (i = 1, 2, ..., N) \tag{2.1c}$$

where X represents the vector of design variables, $f(X)$ is the objective function, $G_j(X)$ is the jth behavior constraint, and x_i^L and x_i^U are the lower and upper limits on the ith design variable, respectively, and J and N denote the number of behavior constraints and design variables, respectively.

Based on the approximations, such as linear, reciprocal, conservative, two-point adaptive nonlinear, multivariate Hermite, *etc.*, the original optimization problem of Eq. (2.1) is changed into a sequence of explicit approximate problems as follows:

$$Minimize \qquad f^{(k)}(X) = \tilde{f}(X) \tag{2.2a}$$

$$Subject \quad to: \qquad G_j^k(X) = \tilde{G}_j(X) \leq 0, \qquad (j = 1, 2, ..., J) \tag{2.2b}$$

$$x_i^L \leq x_i \leq x_i^U \qquad (i = 1, 2, ..., N) \tag{2.2c}$$

where k is the iteration number.

Because of the approximation involved, exact function evaluations are avoided in solving (2.2a) and (2.2b), and computational savings are realized. Only at the convergent solution of the problem (2.2), exact objective and constraint function calculations are needed, and the approximations based on the information of the convergent point are constructed. The process is repeated until convergence.

However, the approximations may result in significant errors if the nonlinearities of the approximation are not closer to those of the original objective and constraint functions. Due to inaccurate approximations, optimization algorithms may be difficult or may never converge without a proper choice of move limits. To avoid this problem, constructing accurate approximations is very important.

In summary, the approximations play an important role in the optimization of large-scale or complex structures. The approximations can reduce the high computational cost required in evaluating objective function and constraints hundreds or thousands of times. Also, the approximations are able to transfer the analysis package into the optimization program when the structural analysis program is large, or if the analyst does not have access to the source code of the program.

2.2 Availability of Gradients and Their Use

An important task in optimal design is to obtain sensitivity derivatives, which are used for studying the effect of parametric modifications, calculating the search directions for finding an optimum design, and constructing function approximations. The calculation of the sensitivity of structural response to changes in design variables is often the major computational cost of the optimization process. For a simple truss problem with n design variables, computing the first-order gradients in terms of all the design variables requires n FE analyses. Computing the gradients of constraint functions will be very expensive if the problem has a large number of

design variables. Also, for some complex problems, the sensitivity analysis may not be easily computed. The approximate gradients information has to be used for optimization.

One commonly used technique for calculating derivatives of response with respect to a design variable is the finite-difference approximation. This technique requires only the function estimations, so it is very useful when the exact derivatives are difficult to determine. However, the finite-difference is often computationally expensive and has accuracy problems.

Use of approximations is an efficient way to obtain the sensitivity derivatives. Once the approximation is constructed, the derivatives can be easily calculated from the explicit functions. If the approximation is closer to the original function at the design point, it provides good sensitivity estimations at the point. Also, no extra exact analyses are needed in computing the derivatives.

2.3 One-point Approximations

In this section, several one-point approximations (linear, reciprocal and conservative) are introduced. One-point approximation means that the approximation is constructed based on the function value and gradients information of one point. Usually, this point is selected as the most current point in the iteration process. The most commonly used approximations of objective and constraint functions are based on one-point information, *i.e.*, the function and its first derivatives at a single design point.

2.3.1 Linear Approximation

The simplest approximation is the linear approximation, which is a first-order Taylor series expansion at a design point X_1:

$$\tilde{g}(X) = g(X_1) + \sum_{i=1}^{n} \frac{\partial g(X_1)}{\partial x_i}(x_i - x_{i,1}) \tag{2.3.1}$$

where x_i is the i^{th} component of variables X and $x_{i,1}$ is the i^{th} component of the known point X_1. This approximation is very popular since the function and its derivatives are needed in search

direction calculation, and no additional computation is involved in developing the approximate function. However, for many applications the linear approximation is inaccurate even for design points X that are close to X_1. Accuracy can be increased by retaining additional terms in the Taylor series expansion, but it requires the costly calculation of higher-order derivatives. A more attractive alternative is to find intervening variables that make the approximate function more linear. One of the popular intervening variables is the reciprocal of x_i, which makes the following reciprocal approximation.

2.3.2 Reciprocal Approximation

The reciprocal approximation is the first-order Taylor series expansion in the reciprocals of the variables $y_i = 1/x_i$ $(i = 1, 2, ..., n)$. It can be written in terms of the original variables x_i:

$$\tilde{g}(X) = g(X_1) + \sum_{i=1}^{n} \frac{\partial g(X_1)}{\partial x_i}(x_i - x_{i,1})(\frac{x_{i,1}}{x_i}) \tag{2.3.2}$$

This approximation has been proven efficient for truss structures with stress and displacement constraints because in statically determinate structures, stress and displacement constraints are linear functions of the reciprocals of the design variables X.

However, there is one problem in the reciprocal approximation given in Eq. (2.3.2). The approximation becomes unbounded when one of the variables approaches zero. A modified approximation was presented by Haftka, et al in Ref. [2], which is

$$\tilde{g}_m(X) = g(X_2) + \sum_{i=1}^{n} \frac{\partial g(X_2)}{\partial x_i}(x_i - x_{i,2})(\frac{x_{mi} + x_{i,2}}{x_{mi} + x_i}) \tag{2.3.3}$$

where X_2 is the current point. The values of x_{mi} were evaluated by matching with the derivatives at the previous point X_1, that is

$$\frac{\partial g(X_1)}{\partial x_i} = (\frac{x_{mi} + x_{i,2}}{x_{mi} + x_{i,1}})^2 \frac{\partial g(X_2)}{\partial x_i} \tag{2.3.4}$$

or

$$x_{mi} = \frac{x_{i,2} - \eta_i x_{i,1}}{\eta_i - 1} \tag{2.3.5}$$

where

$$\eta_i^2 = \frac{\partial g(X_1)}{\partial x_i} \Big/ \frac{\partial g(X_2)}{\partial x_i} \tag{2.3.6}$$

When the ratio of the derivatives is negative, the derivatives at the previous point X_1 are not matched. In that case, x_{mi} is set to a very large number, so that the linear approximation is used for the ith variable.

This modified reciprocal approximation is a two-point approximation because two-point information is used to construct the approximation given in Eq. (2.3.3).

2.3.3 Conservative Approximation

Conservative approximation, as presented by Starnes and Haftka, 1979 in Ref. [3] is a hybrid form of the linear and reciprocal approximations and is more conservative than both. The approximation is given as

$$\tilde{g}(X) = g(X_1) + \sum_{i=1}^{N} C_i \frac{\partial g(X_1)}{\partial x_i}(x_i - x_{i,1}) \tag{2.3.7}$$

where

$$C_i = \begin{cases} \frac{x_{i,1}}{x_i}, & if \quad x_{i,1}\frac{\partial g}{\partial x_i} \leq 0 \\ 1 & otherwise \end{cases} \tag{2.3.8}$$

In the above approximation, $C_i = 1$ corresponds to the linear approximation, and $C_i = x_{i,1}/x_i$ corresponds to the reciprocal approximation.

The conservative approximation is not the only hybrid linear-reciprocal approximation possible. Sometimes physical considerations may dictate the use of linear approximation for some variables and the reciprocal for others. However, as can be easily checked, the conservative approximation has the advantage of being convex. If the objective function and all the constraints are approximated by the conservative approximation, the approximate optimization problem is convex. Convex problems are guaranteed to have only a single optimum, and they are amenable to treatment by dual methods.

2.4 Two-point Adaptive Nonlinear Approximations

In this section, several two-point approximations are introduced. These approximations were presented by Wang and Grandhi in Refs. [4 - 6]. Two-point approximation means that the approximation is constructed based on the function values and gradients information of two points. Usually, one is selected as the most current point and another is the previous point in the iteration process.

Adaptability represents the capability of automatically matching the nonlinearity of various functions. For one-point approximations, the nonlinearity of the approximations is fixed since the intervening variables are fixed. In general, selecting appropriate intervening variables is extremely difficult for different engineering problems. For the stress and displacement constraints of the truss structures, the reciprocal approximation can yield accurate results by using the reciprocals of the design variables. However, these reciprocal intervening variables may not be good for other constraints of truss structures or other structures. For practical engineering problems, the use of fixed intervening variables is difficult to adopt for different constraints, and selection of the intervening variables is also quite difficult and requires tremendous experience and knowledge. Therefore, the use of adaptive approximate models or changeable intervening variables for different types of problems is necessary. The following two-point approximations are capable of adjusting their nonlinearities automatically by using two-point information.

2.4.1 Two-point Adaptive Nonlinear Approximation (TANA) [4]

TANA is a two-point adaptive approximation and was presented by Wang and Grandhi in Ref. [4] using adaptive intervening variables. The intervening variables are defined as

$$y_i = x_i^r, \qquad i = 1, 2, ..., n \tag{2.4.1}$$

where r represents the nonlinearity index, which is different at each iteration, but is the same

for all variables. The nonlinearity index was determined by matching the function value of the previous design point; that is, r is numerically calculated so that the difference of the exact and approximate $g(X)$ at the previous point X_1 becomes zero,

$$g(X_1) - \{g(X_2) + \frac{1}{r}\sum_{i=1}^{n} x_{i,2}^{1-r}\frac{\partial g(X_2)}{\partial x_i}(x_{i,1}^r - x_{i,2}^r)\} = 0 \tag{2.4.2}$$

r can be any positive or negative real number (not equal to zero). The two-point adaptive nonlinear approximation (TANA) is

$$\tilde{g}(X) = g(X_2) + \frac{1}{r}\sum_{i=1}^{n} x_{i,2}^{1-r}\frac{\partial g(X_2)}{\partial x_i}(x_i^r - x_{i,2}^r) \tag{2.4.3}$$

This approximation has been extensively used in truss, frame, plate and turbine blade structural optimization and probabilistic design. The results presented in Refs. [4 - 6] demonstrate the accuracy and adaptive nature of building a nonlinear approximation.

Another two-point approximation, Two-point Exponential Approximation (TPEA) [7], is similar to the above TANA. The TPEA method has N different nonlinear indices, p_i, for each variable and matches the derivatives of exact and approximate function values at the previous point to evaluate p_i, while TANA matches only the function values of exact and approximate calculations at the previous point for finding r. The TPEA method uses the derivative values at two points and the function value at the current point, while TANA uses the function values at two points and the derivative values at the current point. To utilize more information in constructing a better approximation, the TANA1 and TANA2 combine TANA and TPEA and produces improved approximations. In TANA1 and TANA2, both function and derivative values of two points are utilized in developing the approximations.

2.4.2 Improved Two-point Adaptive Nonlinear Approximation(TANA1) [6]

The intervening variables given in Ref. [7] are used, that is,

$$y_i = x_i^{p_i}, \qquad i = 1, 2, ..., n \tag{2.4.4}$$

where p_i is the nonlinear index, which is different for each design variable. The approximate function is assumed as

$$\tilde{g}(X) = g(X_1) + \sum_{i=1}^{n} \frac{\partial g(X_1)}{\partial x_i} \frac{x_{i,1}^{1-p_i}}{p_i}(x_i^{p_i} - x_{i,1}^{p_i}) + \varepsilon_1 \tag{2.4.5}$$

where ε_1 is a constant, representing the residue of the first-order Taylor approximation in terms of the intervening variables y_i ($y_i = x_i^{p_i}$). Unlike the other two-point approximations, this approximation is expanded at the previous point X_1 instead of the current point X_2. The reason is that if the approximation was constructed at X_2, the approximate function value would not be equal to the exact function value at the expanding point because of the correction term ε_1. In actual optimization, to obtain more accurate predictions closer to the current point, X_1 is selected as the expansion point. The approximate function and its derivative values are matched with the current point.

By differentiating Eq. (2.4.5), the derivative of the approximate function with respect to the ith design variable x_i is written as

$$\frac{\partial \tilde{g}(X)}{\partial x_i} = (\frac{x_i}{x_{i,1}})^{p_i - 1} \frac{\partial g(X_1)}{\partial x_i}, \qquad i = 1, 2, ..., n \tag{2.4.6}$$

From this equation, p_i can be evaluated by letting the exact derivatives at X_2 equal the approximation derivatives at this point, that is

$$\frac{\partial g(X_2)}{\partial x_i} = \frac{\partial \tilde{g}(X_2)}{\partial x_i} = (\frac{x_{i,2}}{x_{i,1}})^{p_i - 1} \frac{\partial g(X_1)}{\partial x_i}, \qquad i = 1, 2, ..., n \tag{2.4.7}$$

where p_i can be any positive or negative real number (not equal to zero). Eq. (2.4.7) has n equations and n unknown constants. It is easy to solve because each equation has a single

unknown constant p_i. Here, a simple adaptive search technique is used to solve them. The numerical iteration for calculating each p_i starts from p_i=1. When p_i is increased or decreased by a step length (0.1), the error between the exact and approximation derivatives at X_2 is calculated. If this error is smaller than the initial error (e.g. corresponding to $p_i = 1$), the above iteration is repeated until the allowable error (0.001) or limitation of p_i is reached, and p_i is determined. Otherwise, the step length of p_i is decreased by half, and the above iteration process is repeated until the final p_i is obtained. This search is computationally inexpensive because Eq. (2.4.7) is available in a closed form and is easy to implement.

Eq. (2.4.7) matches only the derivative values of the current point, so a difference between the exact and approximate function values at the current point may exist. This difference is eliminated by adding the correct term, ε_1, in the approximation. ε_1 is computed by matching the approximate and exact function values at the current point:

$$\varepsilon_1 = g(X_2) - \{g(X_1) + \sum_{i=1}^{n} \frac{\partial g(X_1)}{\partial x_i} \frac{x_{i,1}^{1-p_i}}{p_i}(x_{i,2}^{p_i} - x_{i,1}^{p_i})\} \tag{2.4.8}$$

ε_1 is a constant during a particular iteration. This method is simple and more importantly the new approximation function and derivative values are equal to the exact values at the current point.

2.4.3 Improved Two-point Adaptive Nonlinear Approximation (TANA2) [6]

TANA2 uses the intervening variables given in Eq. (2.4.4). The approximation is written by expanding the function at X_2:

$$\tilde{g}(X) = g(X_2) + \sum_{i=1}^{n} \frac{\partial g(X_2)}{\partial x_i} \frac{x_{i,2}^{1-p_i}}{p_i}(x_i^{p_i} - x_{i,2}^{p_i}) + \frac{1}{2}\varepsilon_2 \sum_{i=1}^{n}(x_i^{p_i} - x_{i,2}^{p_i})^2 \tag{2.4.9}$$

This approximation is a second-order Taylor expansion in terms of the intervening variables y_i ($y_i = x_i^{p_i}$), in which the Hessian matrix has only diagonal elements of the same value ε_2. Therefore, this approximation doesn't need the calculation of the second-order derivatives. Unlike the original second-order approximation, this approximation is expanded in terms of the

intervening variables y_i, so the error from the approximate Hessian matrix is partially corrected by adjusting the nonlinearity index p_i. In contrast to the true quadratic approximation, this approximation is closer to the actual function for highly nonlinear problems because of its adaptability. Eq. (2.4.9) has $n+1$ unknown constants, so $n+1$ equations are required. Differentiating Eq. (2.4.9), n equations are obtained by matching the derivatives with the previous point X_1:

$$\frac{\partial g(X_1)}{\partial x_i} = (\frac{x_{i,1}}{x_{i,2}})^{p_i-1}\frac{\partial g(X_2)}{\partial x_i} + \varepsilon_2(x_{i,1}^{p_i} - x_{i,2}^{p_i})x_{i,1}^{p_i-1}p_i \qquad i = 1,2,...,n \tag{2.4.10}$$

Another equation is obtained by matching the exact and approximate function values with the previous point X_1, that is

$$g(X_1) = g(X_2) + \sum_{i=1}^{n}\frac{\partial g(X_2)}{\partial x_i}\frac{x_{i,2}^{1-p_i}}{p_i}(x_{i,1}^{p_i} - x_{i,2}^{p_i}) + \frac{1}{2}\varepsilon_2\sum_{i=1}^{n}(x_{i,1}^{p_i} - x_{i,2}^{p_i})^2 \tag{2.4.11}$$

There are many algorithms for solving these $n+1$ equations as simultaneous equations. Again, a simple adaptive search technique is used. First, ε_2 is fixed at a small initial value (0.5), the numerical iteration described in Section 2.4.2 is used to solve each p_i, and the differences between the exact and approximate function and derivative values at X_1 are calculated. Then, ε_2 is increased or decreased by a step length (0.1), p_i, and the differences between the exact and approximate function and derivative values at X_1 are recalculated. If these differences are smaller than the initial error (e.g. corresponding to $\varepsilon_2 = 0.5$), the iteration is repeated until the allowable error (0.001) or limitation of ε_2 is reached, and the optimum combination of ε_2 and p_i is determined.

In the TANA2 method, the exact function and derivative values are equal to the approximate function and derivative values, respectively, at both points. Therefore, this approximation is more accurate than others.

2.5 Multi-point Hermite Approximation

In the literature, the Hermite interpolation scheme is presented for a single variable by using

multiple data points. In this work, this concept is extended for multi-dimensional problems. The advantage of using Hermite interpolation is that it makes use of both function and derivative information in building the approximation. The interpolating polynomial retains the same function and derivative values as the original information at each of the known data points. The Hermite p-point formula gives accurate results when the function to be approximated is identified with any polynomial of degree not exceeding $2p-1$ [8]. This approximation can be applied for large scale problems with hundreds of design variables because the approximation is constructed only by simple algebraic calculations. Mathematical details of this approximation are described below.

For a univariate function $f(x)$, assuming that the values of the function and first-order derivatives at p different points x_i $(i = 1, 2, ..., p)$ are $y_i = f(x_i)$ and $y_i' = f'(x_i)$, the Hermite interpolation formula can be given as in Ref. [8]:

$$\tilde{f}(x) = \sum_{i=1}^{p} \{y_i + [y_i' - 2y_i h_i'(x_i)](x - x_i)\} h_i^2(x) \tag{2.5.1a}$$

where

$$h_i(x) = \prod_{j=1, j\neq i}^{p} \frac{x - x_j}{x_i - x_j} \tag{2.5.1b}$$

$$h_i'(x_i) = \sum_{j=1, j\neq i}^{p} \frac{1}{x_i - x_j} \tag{2.5.1c}$$

At each point of x_i (i=1,2,...,p), $\tilde{f}(x)$ satisfies

$$\tilde{f}(x_i) = y_i \tag{2.5.2a}$$

$$\tilde{f}'(x_i) = y_i' \tag{2.5.2b}$$

To construct a multivariate Hermite approximation, the above Eqs. (2.5.1b) and (2.5.1c) can be extended for n variables as

$$h_i(S) = \prod_{j=1, j\neq i}^{p} \frac{(S - S_j)^T (S_i - S_j)}{(S_i - S_j)^T (S_i - S_j)} \tag{2.5.3a}$$

$$\nabla h_i(S) = [\frac{\partial h_i(S)}{\partial s_1}, \frac{\partial h_i(S)}{\partial s_2}, ..., \frac{\partial h_i(S)}{\partial s_n}]^T \tag{2.5.3b}$$

$$\frac{\partial h_i(S)}{\partial s_k} = h_i(S) \sum_{j=1, j\neq i}^{p} \frac{s_{k,i} - s_{k,j}}{(S - S_j)^T (S_i - S_j)} \qquad k = 1, 2, ..., n \tag{2.5.3c}$$

where S is a vector of $\{s_1, s_2, ..., s_n\}$, which can represent original variables or intervening variables, n is the number of variables, and $s_{k,i}$ represents k^{th} variable of the ith data point.

Using Eqs. (2.5.3a), (2.5.3b) and (2.5.3c), the one-dimensional Hermite interpolation given in Eq.(2.5.1a) can be extended as the following multivariate Hermite expression:

$$\tilde{f}(S) = \sum_{i=1}^{p} \{g(S_i) + [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)]^T (S - S_i)\} h_i^2(S) \tag{2.5.4}$$

where $g(S_i)$ and $\nabla g(S_i)$ are the function value and gradient vector at the ith known point, respectively. $\nabla h_i(S_i)$ is obtained by substituting S_i into Eq. (2.5.3c), in which $h_i(S_i)$ becomes 1 (proof is given in Appendix A).

Differentiating Eq. (2.5.4), a gradient formula of the function approximation can be obtained as

$$\begin{aligned} \nabla \tilde{f}(S) \;=\; & \sum_{i=1}^{p} \Big\{ 2h_i(S) \nabla h_i(S) \{g(S_i) + [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)]^T (S - S_i)\} \\ & + h_i^2(S) [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)] \Big\} \end{aligned} \tag{2.5.5}$$

Unlike the two-point Hermite approximation presented in [2], the multivariate Hermite formula given in Eq. (2.5.4) reduces to the one-dimensional Hermite interpolation given in Eq.(2.5.1a) when $n = 1$. Furthermore, Eq. (2.5.4) can be used for multiple point approximation, not just limited to only two points. The two-point Hermite approximation in [2] is based on projection of a point onto the line connecting the two known points. The function $f(X)$ is first approximated by a cubic Hermite polynomial at the projection point, and then linearly extrapolated to the test points. It can be used only for two-point approximations because the cubic polynomial is derived from two-point Hermite formula of Eq. (2.5.1a). The present

p-point n-dimensional Hermite formula is an explicit function with a $2p-1$ polynomial degree, which possesses the same value and the same derivatives at each of the p data points as the exact function. The demonstration of this characteristic is given in Appendix A. The nonlinearity of the Hermite approximation will change as the number (p) of the known points increases because it is a function with a $2p-1$ polynomial degree. In order to control the nonlinearity and make the constructed approximations closer to the actual nonlinear functions, two types of intervening variables are given below.

2.5.1 Preselected Intervening Variables

Usually, fixed types of intervening variables are selected in terms of element properties of the actual problem. For example, for truss structures with stress and displacement constraints, the intervening variables can be assumed as the reciprocal of the physical variables so that the behavior functions are fairly linear. For frame structures with displacement constraints, the reciprocal section properties can be selected as the intermediate variables [9]. It has been demonstrated that the function approximations obtained by selecting appropriate intermediate design variables and then expanding using the Taylor series are closer to the actual constraints for some of the structural elements.

The preselected intervening variables can be written as

$$s_k = T_k(X) \qquad k = 1, 2...n \tag{2.5.6}$$

in which $T_k(X)$ is a function of original variables, X. For truss and frame structures, $T_k(X) = x_k^r$ is an example. The value r represents the preselected nonlinearity in terms of variables X. For the above mentioned structures, $r = -1$ is often used in the Taylor series expansion. For the present Hermite approximation, the value of r can be controlled by

$$(2p-1)r = r_0 \tag{2.5.7}$$

where r_0 represents the actual nonlinearity in terms of variables $T(X)$. Eq. (2.5.7) makes

the nonlinearity of the constructed approximation have the same nonlinearity as the actual function.

Many researchers have shown that the selection of appropriate intervening variables improves the quality of approximations; however, they require experience and knowledge. Often finding intervening variables is extremely difficult for some complex multidisciplinary problems. Therefore, use of an automated adaptive intervening variables calculation for certain classes of problems such as aeroservoelasticity, structural control, probabilistic analysis, *etc.* is necessary where the relationships are not transparent.

2.5.2 Adaptive Intervening Variables

Adaptive intervening variables are denoted as $S = (s_1, s_2, ..., s_n)^T$.

$$s_k = x_k^r \qquad k = 1, 2, ..., n \tag{2.5.8}$$

where x_k $(k = 1, 2, ..., n)$ are the original design variables, and r represents the nonlinearity index, which is different at each iteration, but the same for all variables. In order to determine this index, a feedback formula based on multiple point information is established as follows. Let the value of approximate function $\tilde{f}(S)$ given in Eq. (2.5.4) at the remaining one point (a known point except p of the earlier used points), for example Y equal the value of the exact function $g(Y)$ at this point, that is

$$\tilde{f}(S(Y)) = g(Y) \tag{2.5.9}$$

where Y is a selected point for comparing Hermite approximation and computing the r value. The most recent data point is used for computing the r value so that the function behavior around the current design vector is represented. Using Equations (2.5.3), (2.5.4) and (2.5.8), the nonlinearity index can be obtained from the following multi-point feedback formula:

$$g(S(Y)) - \Big\{\sum_{i=1}^{p}\{g(S(X_i)) + \frac{1}{r}\sum_{k=1}^{n} x_{k,i}^{1-r}[\frac{\partial g(S(X_i))}{\partial x_k} - 2g(X_i)\frac{\partial h_i(S(X_i))}{\partial x_k}] \cdot (y_k^r - x_{k,i}^r)\}$$

$$h_i^2(S(Y))\Big\} = 0 \tag{2.5.10}$$

where $\partial h_i(S(X_i))/\partial x_k$ can be obtained from

$$\frac{\partial h_i(S)}{\partial x_k} = r x_k^{r-1} \frac{\partial h_i(S)}{\partial s_k} \tag{2.5.11}$$

The nonlinearity index r is numerically solved. Equation (2.5.10) may result in more than one root, either positive or negative. The r value closest to unity is taken as the root because the intervening variables resemble the original variables, and no additional effort is needed for finding the intervening variables.

Similarly, the derivatives of $\tilde{f}(S)$ with respect to original variables x_k can be obtained as

$$\frac{\partial f(S)}{\partial x_k} = r x_k^{r-1} \frac{f(S)}{\partial s_k} \tag{2.5.12}$$

2.6 Approximation Comparisons

Several examples are selected to compare the accuracy of the approximations introduced above. The relative and absolute errors are calculated as follows:

$$Relative \quad Error = \frac{Exact - Approximation}{Exact} \tag{2.6.1a}$$

$$Absolute \quad Error = Exact - Approximation \tag{2.6.1b}$$

The examples include explicit and implicit constraint functions. The constraint function of the 313-member frame structure requires a finite element analysis. In all of the examples, the test points are derived using

$$X = X_0 + \alpha_s D \tag{2.6.2}$$

where X_0 is an initial point, which is defined as the expanding point for all the approximations except TANA1 and Hermite. Instead, it is defined as the matching point for TANA1 and Hermite. α_s is a step length, and D is a direction vector which is selected as $D = \{1, 1, 1, 1...\}^T$

for case 1, $D = \{-1, 1, -1, 1, ...\}^T$ for case 2, $D = \{1, 0, 1, 0, ...\}^T$ for case 3, and $D = \{0, 1, 0, 1, ...\}^T$ for case 4.

Example 2.1

This example is taken from Ref. [10], and the constraint function is defined as

$$g(X) = \frac{10}{x_1} + \frac{30}{x_1^3} + \frac{15}{x_2} + \frac{2}{x_2^3} + \frac{25}{x_3} + \frac{108}{x_3^3} + \frac{40}{x_4} + \frac{47}{x_4^3} - 1.0$$

All of the approximations except TANA1 and Herimite are expanded at the point $X_2(1, 1, 1, 1)$, and the previous point is selected as $X_1(1.2, 1.2, 1.2, 1.2)$. TANA1 is expanded at X_1 and matched with the values at X_2. The nonlinearity index r for the Hermite approximation with adaptive intervening variables is determined by matching the function value at X_2. The two-point Hermite approximation is based on $X(0.5, 0.5, 0.5, 0.5)$ and $X(1.2, 1.2, 1.2, 1.2)$. The three-point Hermite approximation is based on $X(0.5, 0.5, 0.5, 0.5)$, $X(0.75, 0.75, 0.75, 0.75)$ and $X(1.2, 1.2, 1.2, 1.2)$. The relative errors of several methods for four cases are plotted in Figs. 2.1a, 2.1b, 2.1c and 2.1d for four cases. Fig. 2.1a shows that when design variables are changed along the same direction as X_1 and X_2 points (case 1), the two-point and three-point approximations have the best accuracy (the relative errors are almost zero everywhere). TANA-2 also has very good accuracy, in which the absolute values of relative errors are smaller than 7% everywhere. The errors of TANA-1 and TANA are smaller than 20% everywhere. TANA-2, TANA-1 and TANA2 have lower errors (almost zero) when $\alpha_s > 0$ because the second point lies on the right side of the α_s axis. For the other three cases, the second point X_1 does not lie in the same direction as D. When design variables x_1 and x_3 are changed along an opposite direction of x_2 and x_4 (case 2), TANA2 has the smallest errors when $\alpha_s > 0$, while TANA has the best accuracy when $\alpha_s < 0$. TANA1 also works well and has the same results where the errors are smaller than 9% everywhere. The two-point and three-point Hermite approximations are better than the linear and reciprocal approximations, but they are not as good as TANA-2, TANA-1 and TANA. When the design variables are changed only along x_1 and x_3 (case 3), TANA2 has

very small errors (smaller than 4%), and TANA1 has almost the same accuracy as TANA2. The two-point and three-point Hermite approximations are good when $\alpha_s > 0$. When the design variables are changed only along x_2 and x_4 (case 4), TANA2 and TANA1 are accurate when $\alpha_s > 0$, and TANA has the lowest errors when $\alpha_s < 0$. The two-point and three-point Hermite approximations are good when $\alpha_s > 0$, but they don't provide much improvement compared with the linear and reciprocal approximations. For all four cases, the relative errors of linear and reciprocal approximations are large. TANA has a single nonlinearity index r which is equal to -2.7. The nonlinearity indices for TANA1 ,p_i, are -2.7625, -1.5,-2.825,-2.4875 for x_1, x_2, x_3, and x_4 and ε_1 is -0.0862. The nonlinearity indices for x_1, x_2, x_3 and x_4 in TANA2 method are -2.7375, -1.45, -2.825 and -2.475, respectively, and ε_2 is 0.5527. The nonlinearity indices r for the two-point and three-point Hermite approximations are -2.469, -1.436, respectively.

Example 2.2

The three-bar truss example shown in Figure 2.2 is taken from Ref. [11]. The truss is designed subject to stress and displacement constraints with cross-sectional areas A_A, A_B, and A_C ($A_A = A_C$) as design variables. The approximations of a member C stress constraint are examined. The stress constraint using normalized variables is written as

$$g(X) = 1 + \frac{\sqrt{3}}{3x_1} - \frac{2}{x_2 + 0.25x_1}$$

$g(X)$ is expanded at the point $X_0(1.0, 1.0)$ for g_l, g_r, TANA, TANA-1 and TANA-2 approximations. The nonlinearity index r, for two-point, three-point and four-point Hermite approximations with adaptive intervening variables, is determined by matching the function value at this point. The point $X_1(1.5, 1.5)$ is the second point for TANA, TANA-1 and TANA-2 to calculate the nonlinearity index r_1 and p_i. The two-point, three-point and four-point approximations are constructed based on selected data points which are similar to the points generated during the optimization process. The results comparison is shown in Table 2.1, which shows that the Hermite approximations have smaller errors, and TANA, TANA-1 and TANA-

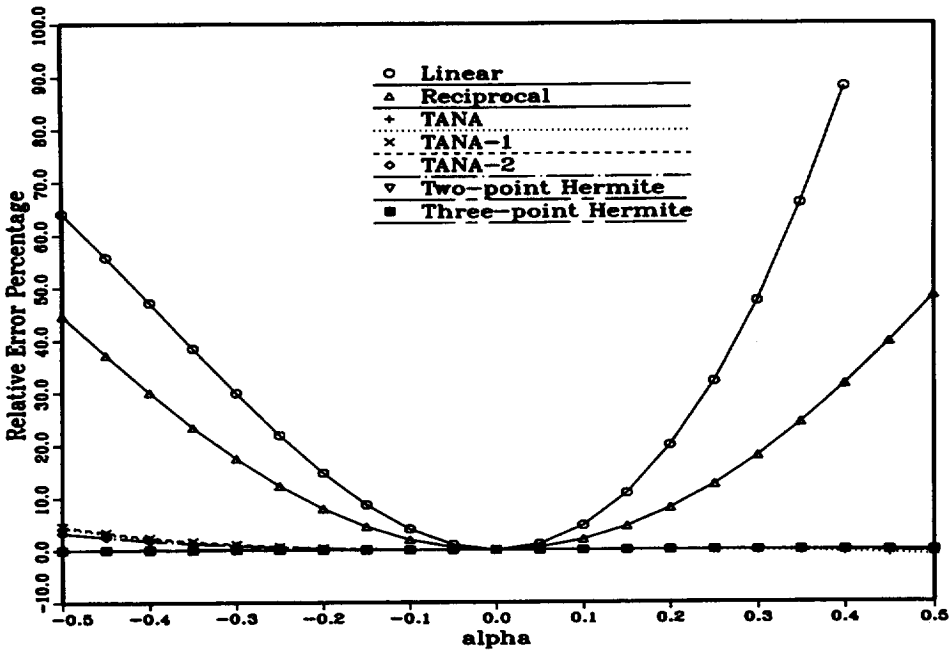


Fig. 2.1a Example 2.1 (Case 1)

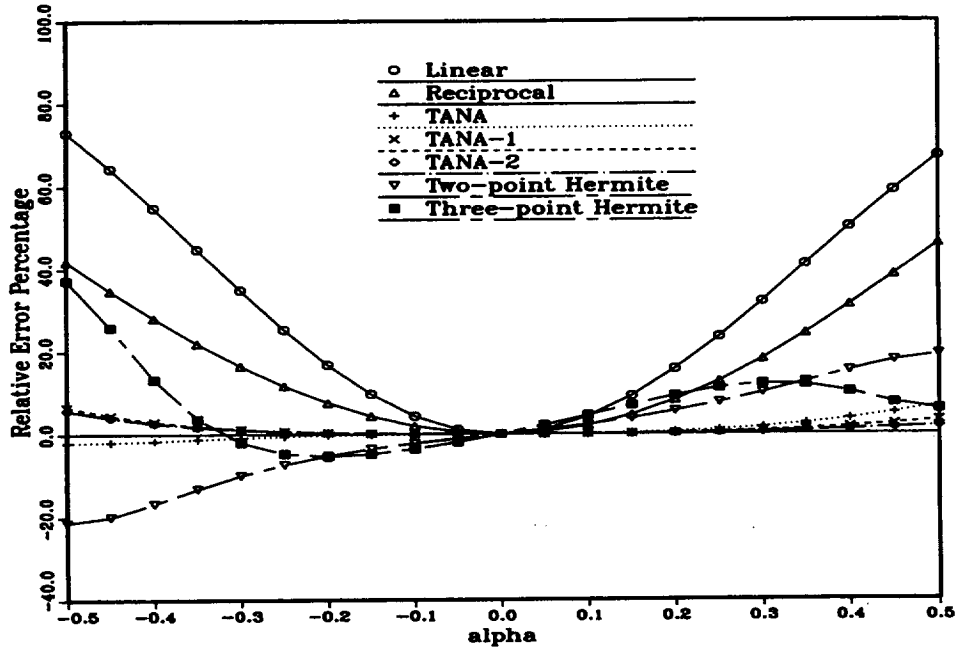


Fig. 2.1b Example 2.1 (Case 2)

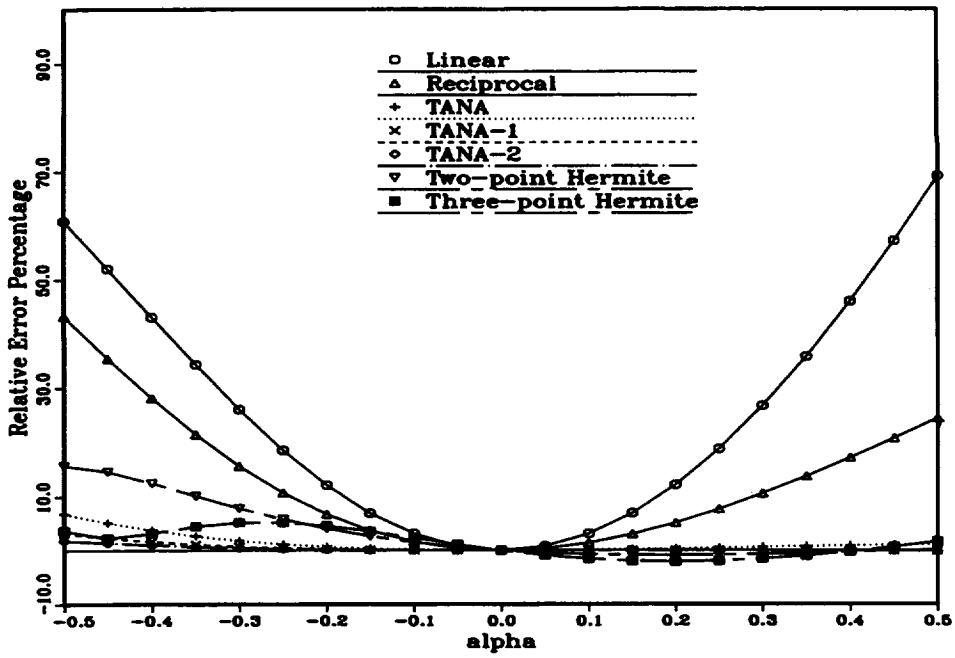


Fig. 2.1c Example 2.1 (Case 3)

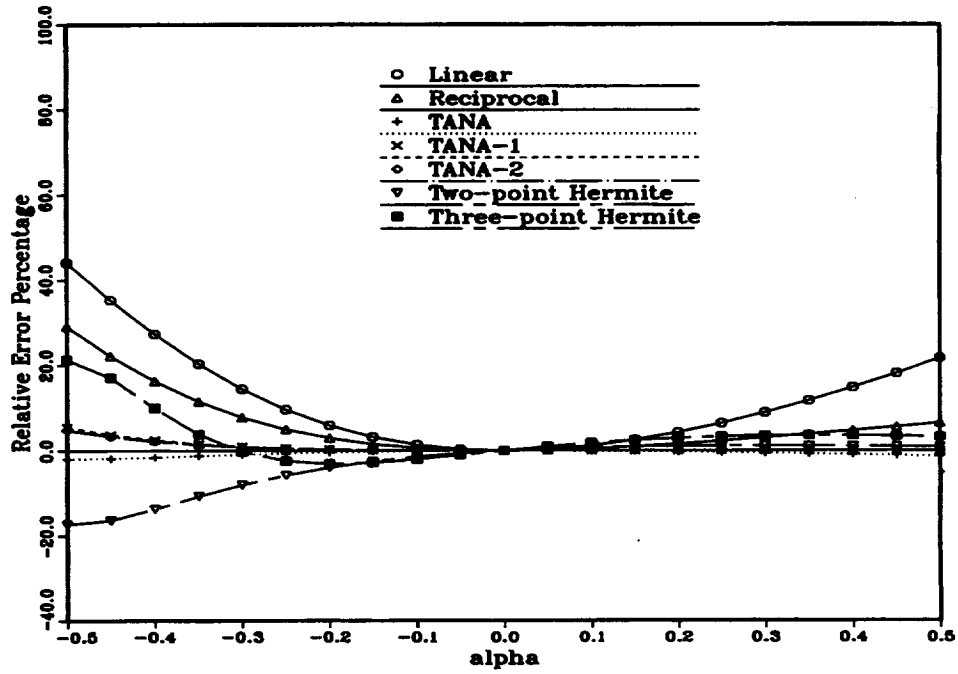


Fig. 2.1d Example 2.1 (Case 4)

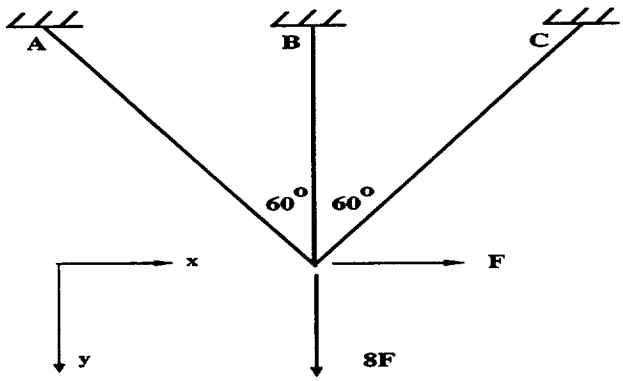


Fig. 2.2 Three–bar Truss

2 have the same accuracy as the reciprocal approximation because the nonlinearity index is equal to -1. Comparing the results of two types of intervening variables, the results of type II intervening variables are again better than those of type I for most of the points.

Example 2.3

This example has an implicit constraint function requiring a finite element analysis. The frame structure shown in Fig. 2.3 is modelled with 313 beam elements with I-sections. The cross-sectional areas of all members are selected as the design variables. The vertical loads at nodes 15, 16, 88, 89 are -26, -30,-18,-20 *kips*, respectively; the horizontal loads at nodes 6, 11, 17 through 65 by 3, 68 through 82 by 7, and 90 through 175 by 5 are 4 *kips*; and the horizontal load at node 1 is 2 *kips*. The approximation to the vertical displacement d at the tip point

Table 2.1 Comparison of Absolute Errors for Example 2.2

x_1	x_2	Linear	Reciprocal	TANA, TANA-1 & TANA-2	2-point Hermite $X_1(1.5, 1.5)$ $X_2(1.25, 0.75)$	3-point Hermite $X_1(1.5, 1.5)$ $X_2(1.25, 0.75)$ $X_3(0.75, 1.0)$	4-point Hermite $X_1(1.5, 1.5)$ $X_2(1.25, 0.75)$ $X_3(1.2, 1.2)$ $X_4(0.75, 1.0)$
0.75	0.75	-0.0852	0.0000	0.0000	0.1456 0.0692*	0.1613 −0.0267*	0.1351 0.1031*
1.00	0.75	-0.0800	0.0267	0.0267	0.0247 0.0133*	0.0092 −0.0510*	-0.0555 −0.0789*
1.25	0.75	-0.0135	0.0803	0.0803	0.0000 0.0000*	0.0000 0.0000*	0.0000 0.0000*
0.75	1.00	0.0439	0.0225	0.0225	0.1775 0.2301*	0.0000 0.0000*	0.0000 0.0000*
1.25	1.00	0.0251	0.0122	0.0122	0.0229 0.0145*	0.0820 0.0224*	0.0006 −0.0215*
0.75	1.25	0.0168	0.0594	0.0594	0.1272 0.2695*	0.0647 −0.0221*	0.0249 0.0302*
1.00	1.25	-0.0533	0.0107	0.0107	0.0348 0.0986*	0.0135 0.1218*	-0.0013 0.0205*
1.25	1.50	-0.1946	0.0059	0.0059	0.0040 0.0154*	0.0032 0.0558*	0.0391 0.0204*
1.50	1.25	-0.0145	0.0066	0.0066	0.0076 −0.0041*	0.0383 −0.0016*	-0.0259 −0.0159*

1. The values with an asterisk (*) represent the preselected intervening variables.
2. The values with an underline are the smallest errors at each test point.

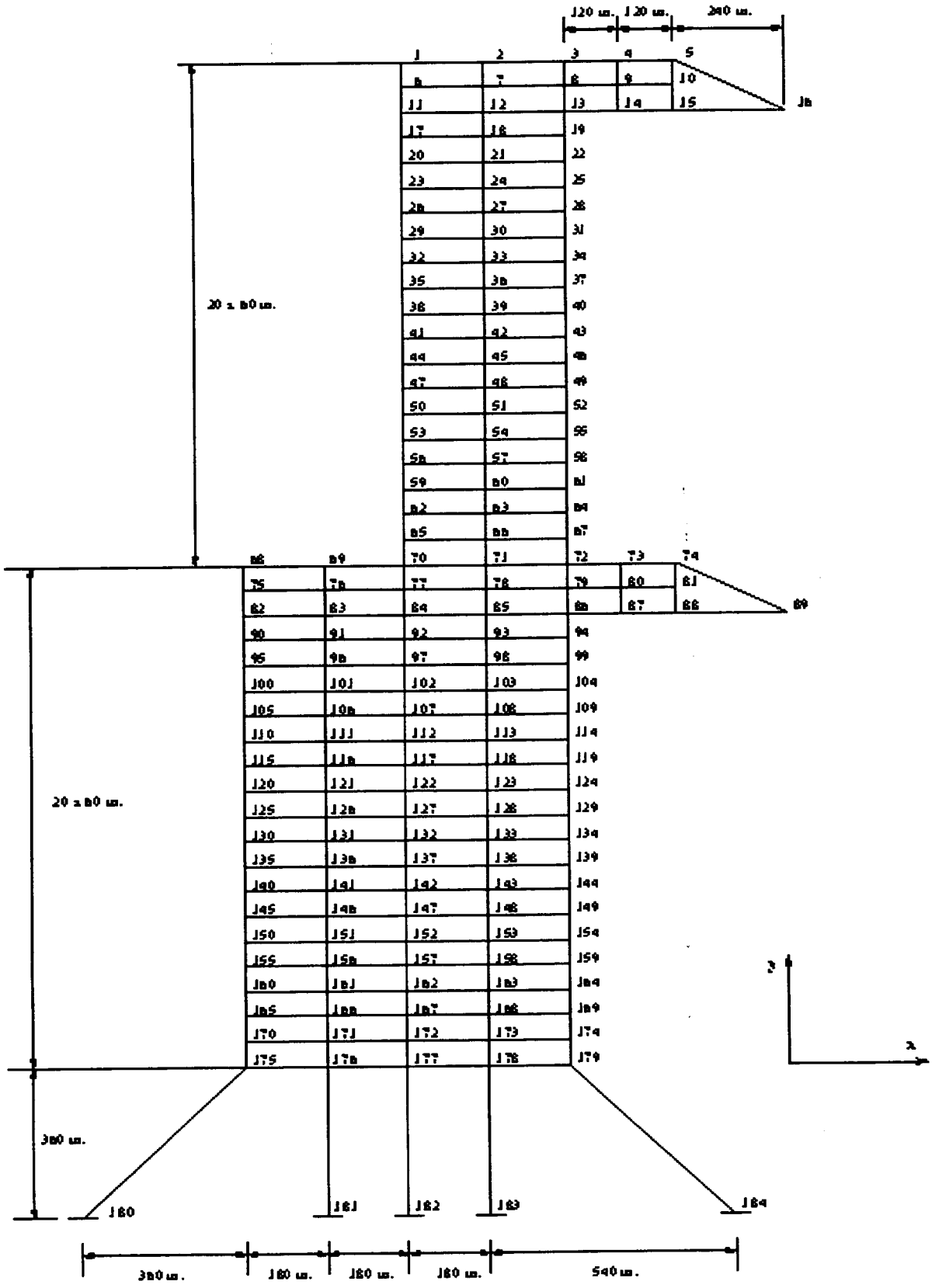


Fig. 2.3 313 member frame

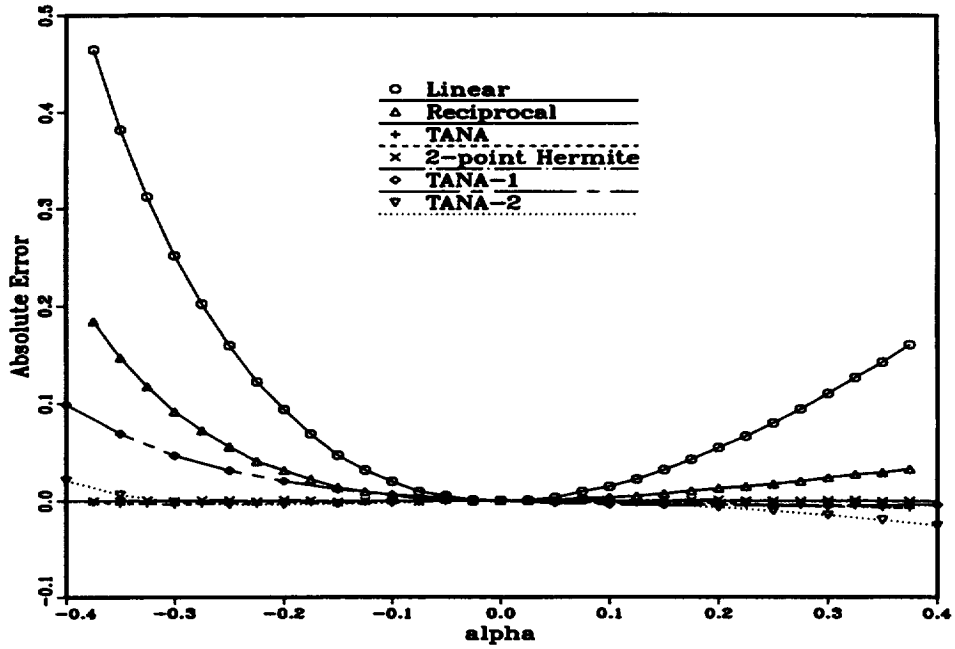


Fig. 2.4a 313 Member Frame (Case 1)

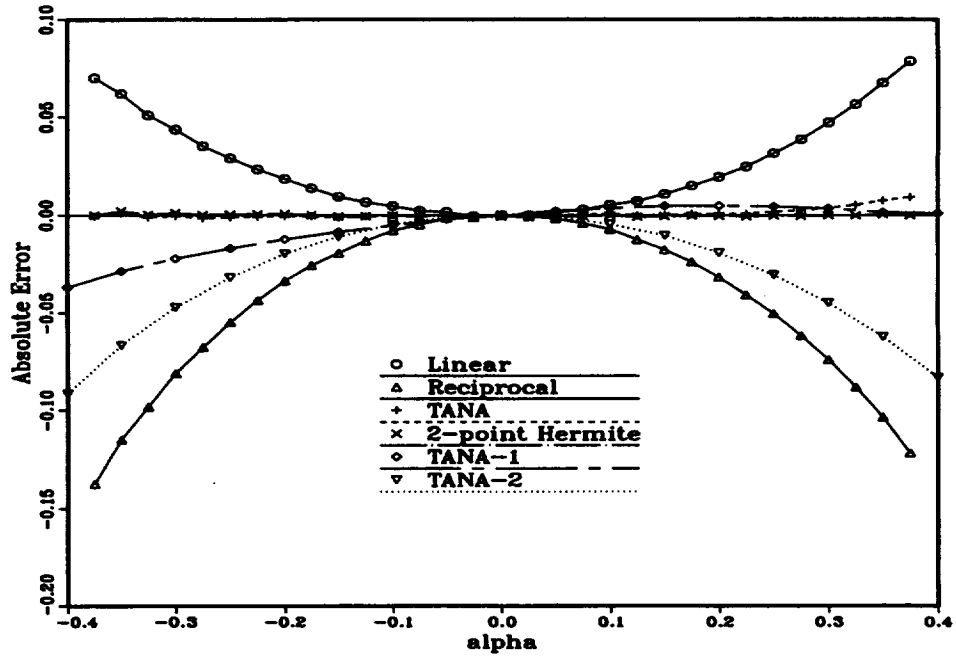


Fig. 2.4b 313 Member Frame (Case 2)

(node 16) is examined, which is an implicit function of 313 design variables and is written as

$$g(X) = d/d_{lim} - 1$$

where d_{lim} is the displacement limit of 4.0 in. Young's modulus is 2.9×10^7 psi, and the Possion's ratio is 0.3. The initial cross-sectional areas X_0 are 30.59 in^2, and the normalized design variables are defined as

$$\tilde{X} = \frac{X}{X_0}$$

All of the approximations except TANA1 and Hermite are expanded at the point $X_2(1, 1,..., 1)$. The previous point is selected as X_1(1.25, 1.25,...,1.25) for case 1 and X_1(1.2,0.8,1.2,0.8,...,1.2) for case 2. TANA1 is expanded at X_1 and matched at X_2 for two cases. The absolute errors of all the approximations are calculated using Eq. (2.6.1b) to avoid the numerical problems in computing the relative errors when the exact $g(X)$ values are close to zero. The results comparison of several methods is shown in Figs. 2.4a and 2.4b. Fig. 2.4a shows that when the design variables are changed along the same direction, the errors of TANA TANA-2 and two-point Hermite are almost zero. TANA-1 has good accuracy when α_s is greater than zero. Fig. 2.4b shows that when the design variables are changed along $D = \{-1, 1\ , -1, 1,...-1\}^T$ (case 2), TANA and two-point Hermite have the lowest errors, and TANA1 and TANA2 also have good accuracy; they are much better than the linear and reciprocal methods. The results indicate that the proposed approximations have very good accuracy even for large scale problems with hundreds of design variables.

2.7 Summary of Approximations

Multi-point function approximations are developed for the mathematical optimization and probability analysis of structures. In the two-point approximations (TANA, TANA-1 and TANA-2), two-point information is used to construct the approximations. In particular, TANA-1 and TANA-2 approximations use both function values and gradients at two points to construct

the approximations. The intervening variables with adaptive nonlinearity index/indices are used to make the approximations closer to the actual nonlinear functions. TANA reproduces exact function values at two points and exact gradients at the current point. TANA-1 reproduces exact gradients at two points and exact function values at the current point. TANA-2 reproduces exact function values and gradients at both points. The Hermite interpolation concept available for a single variable is extended to multidimensional problems by retaining the property of the same function and gradient values at the known data points. Two types of intervening variables are used along with the Hermite approximation for demonstrating the advantages of having adaptive intervening design variables. These intervening variables attempt to make the approximations closer to the actual nonlinear functions.

In the first example with high nonlinearity, 2-point and 3-point Hermite approximations produce the best results as design variables are changed along the same direction for case 1. However, in the other three cases, 2-point and 3-point Hermite approximations are not better than TANA, TANA-1 and TANA-2. The Hermite approximations, particularly for 3-point approximations, slightly oscillate around the known points so that some local points of the approximations are generated. In the last two structural examples, TANA-2 and the Hermite approximations produce very good results.

The computational results indicate that the Hermite approximation can provide very accurate results when the test points are closer to the known data points or when they interpolate. However, for some cases, it may result in an approximation with multiple local points. TANA1, TANA2 and TANA can provide better accuracy than the linear and reciprocal methods for highly nonlinear problems where the functional dependency on design variables is difficult to predict. TANA2 is better than TANA and TANA1 for most cases. For some cases, TANA-2 could be slightly less accurate than the Hermite approximation, but it is much more stable than the Hermite approximation. The accuracy improvement of TANA1 compared to TANA2 is not clearly evident, and it is better than TANA for only a few cases. TANA works

very well for the complex problems (the last two examples) even with a single nonlinearity index. The TANA, TANA-1 and TANA-2 approximations perform extremely well in extrapolating a function. These adaptive two point approximations are highly effective for large scale structures.

*

References

[1] Schmit, L. A. and Farshi, B., "Some Approximation Concepts for Structural Synthesis", AIAA Journal, Vol. 12, No. 5, 1974, pp. 692-699.

[2] Haftka, R. T., Nachlas, J. A., Watson, L. T., Rizzo, T., and Desai, R., "Two-point Constraint Approximation in Structural Optimization", Computer Methods in Applied Mechanics and Engineering, Vol. 60, No. 3, 1987, pp. 289-301.

[3] Starnes, J. H. and Haftka, R. T., "Preliminary design of composite wings for buckling, stress and displacement constraints", AIAA Journal of Aircraft, Vol. 16, 1979, pp. 564-570.

[4] Wang, L. P. and Grandhi, R. V., "Efficient Safety Index Calculation for Structural Reliability Analysis", Computers and Structures, Vol. 52, No. 1, 1994, pp. 103-111.

[5] Wang, L. P., Grandhi, R. V., and Hopkins, D. A.,"Structural Reliability Optimization Using An Efficient Safety Index Calculation Procedure", International Journal for Numerical Methods in Engineering, Vol. 38, 1995, pp. 1721-1738.

[6] Wang, L. P. and Grandhi, R. V., "Improved Two-point Function Approximation for Design Optimization", AIAA Journal, Vol. 32, No. 9, 1995, pp. 1720-1727.

[7] Fadel, G. M., Riley, M. F. and Barthelemy, J. F. M., "Two-point Exponential Approximation Method for Structural Optimization", Structural Optimization, Vol. 2, No. 2, 1990, pp. 117-124.

[8] F. B. Hildebrand, Introduction to Numerical Analysis, McGraw-Hill, New York, 1974.

[9] W. C. Mills-Curran, R. V. Lust and L. A. Schmit, "Approximation Methods for Space Frame Synthesis", AIAA Journal, Vol. 21, 1983, pp. 1571-1580.

[10] Venkayya, V. B. and Tischler, V. A., "A Compound Scaling Algorithm for Mathematical Optimization", Air Force Wright Aeronautical Labs., WRDC-TR-89-3040, Wright-Patterson AFB, OH, 1989.

[11] Haftka, R. T. and Gurdal, Z. Elements of Structural Optimization, Kluwer, Netherlands, 1992.

CHAPTER 3. SAFETY INDEX AND MPP CALCULATIONS

3.1 Limit State Function

If a structure or part of a structure exceeding a specific limit is unable to perform a required performance, this specific limit is called a limit state. There are two kinds of limit states: one is the load-bearing capacity, and the other is the normal performance. The former mainly concerns the structural safety, and the latter concerns the structural applicability and durability. The structure will be considered unreliable if the failure probability of the structural limit state exceeds the required value.

Assuming that the reliability of the structure depends upon n independent random variables, $X = \{x_1, x_2, ..., x_n\}^T$, the state function of the structural reliability is

$$g(X) = g(x_1, x_2, ..., x_n) \tag{3.1.1}$$

Let

$$g(X) = g(x_1, x_2, ..., x_n) = 0 \tag{3.1.2}$$

Eq. (3.1.2) is the limit state function of the structural reliability, which separates the design space into 'failure' and 'safe' regions, *i.e.*,

$$g(X) > 0, \qquad x_i \in \text{Safe region} \tag{3.1.3a}$$

$$g(X) = 0, \qquad x_i \in \text{Failure surface} \tag{3.1.3b}$$

$$g(X) < 0, \qquad x_i \in \text{Failure region} \tag{3.1.3c}$$

The simplest example of the limit state function can be given as the following stress-strength problem:

$$g(R, S) = R - S = 0 \tag{3.1.4}$$

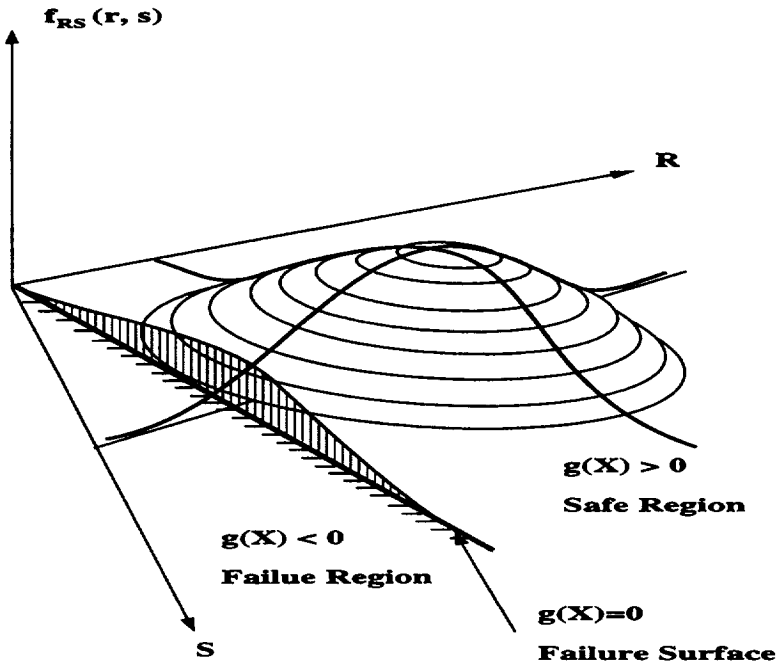


Fig. 3.1 Limit State Surface between Failure and Safe Regions

where R is the strength, S is the stress resultant, and $g(R, S)$ is the limit state function of the structural reliability. Fig. 3.1 shows that the limit state of the line, R=S, separates the design space into two regions: one is the safe region of $S < R$, and the other is the failure region of $S > R$.

For most of structures, the limit states can be divided into three categories:

1. *Ultimate limit states* are related to a structural collapse of part or all of the structure. Examples of the most common ultimate limit states are corrosion, fatigue, deterioration, fire, plastic mechanism, progressive collapse, fracture, etc. Such a limit state should have a very low probability of occurrence, since it may lead to loss of life and major financial losses.

2. *Damage limit states* are related to the damage of the structure. Examples of damage limit states are excessive or premature cracking, deformation or permanent inelastic deformation, etc. Such a limit state is often included in the above category.

3. *Serviceability limit states* are related to disruption of the normal use of the structures.

Examples of serviceability limit states are excessive deflection, excessive vibration, drainage, leakage, local damage, etc. Since there is less danger than in the case of the ultimate limit states, a higher probability of occurrence may be tolerated in such limit states.

3.2 Reliability Index / Safety Index

For simplicity, the limit state function given in Eq. (3.1.4) is taken as an example again. The probability of failure is computed as

$$P_f = \int_{\Omega} f_{RS}(R,S)dRdS \tag{3.2.1}$$

where $f_{RS}(.)$ is the joint probability density function of R and S. Ω is the failure region modeled by the limit state function, $g(X) \leq 0$, as shown in Fig. 3.1.

Assuming that the strength (R) and stress (S) are random variables normally distributed, the limit state function $g(R,S)$ is normally distributed. According to Eqs. (B.4) and (B.6) given in Appendix B, the mean value and standard deviation of the function $g(R,S)$ are given as

$$\mu_g = \mu_R - \mu_S \tag{3.2.2}$$

$$\sigma_g = \sqrt{\sigma_R^2 + \sigma_S^2} \tag{3.2.3}$$

where μ_R and μ_S are the means of R and S, respectively, and σ_R and σ_S are the standard deviations of R and S, respectively. The probability density function of $g(R,S)$ is

$$f_g(g) = \frac{1}{\sigma_g\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{g-\mu_g}{\sigma_g})^2] \tag{3.2.4}$$

The failure probability is

$$P_f = \int_{-\infty}^{0} f_g(g)dg \tag{3.2.5}$$

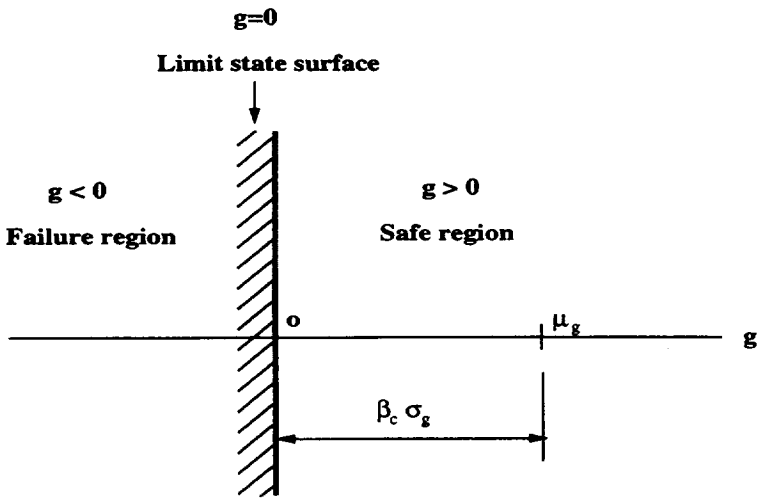


Fig. 3.2 Geometrical Illustration of the Cornell Reliability Index, β_c, (I)

and the reliability or probability is

$$P = 1 - P_f = \int_0^\infty f_g(g)dg \tag{3.2.6}$$

By introducing the standard normalized variable u with

$$u = \frac{g - \mu_g}{\sigma_g} \tag{3.2.7}$$

when $g = 0$, the lower limit of u is given as

$$u_L = \frac{0 - \mu_g}{\sigma_g} = -\frac{\mu_R - \mu_S}{\sqrt{\sigma_R^2 + \sigma_S^2}} \tag{3.2.8}$$

and when $g = +\infty$, the upper limit of u tends to be $+\infty$. When using integral transformation by considering Eqs. (3.2.4) and (3.2.7), as well as the lower and upper limits of u, Eq.(3.2.6) becomes

$$\begin{aligned} P &= \int_{u_L}^{\infty} \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}u^2)du \\ &= 1 - \Phi(-\frac{\mu_R - \mu_S}{\sqrt{\sigma_R^2 + \sigma_S^2}}) \\ &= 1 - \Phi(-\beta) \\ &= \Phi(\beta) \end{aligned} \tag{3.2.9}$$

where $\Phi(.)$ is the standard normal cumulative distribution function. β is defined as the reliability index or safety index for uncorrelated (independent) variables:

$$\beta = \frac{\mu_g}{\sigma_g} = \frac{\mu_R - \mu_S}{\sqrt{\sigma_R^2 + \sigma_S^2}} \tag{3.2.10}$$

β is the reciprocal of the estimate of coefficient of variance in $g(R, S)$. If the difference between the means of the strength and the stress is reduced, β decreases and then P_f increases. Otherwise, if either σ_R or σ_S or both are increased, β will become smaller and hence P_f will increase. Therefore, β is a direct measure of the reliability of the structure, and a larger β represents greater reliability or lower probability of failure.

The safety index given in Eq. (3.2.10) is also called the Cornell reliability index, β_c, [1]. The geometrical illustration of the Cornell reliability index is shown in Fig. 3.2. In this one-dimensional case, the failure surface is simply a point $g = 0$. The idea behind the reliability index definition is that the distance from location measure μ_g to the limit state surface provides a good measure of reliability. The distance is measured in units of the uncertainty scale parameter σ_g. Fig. 3.3 gives further geometrical illustration of the safety index, which shows that the shaded area to the left of the origin is the probability of failure.

The above one-dimensional geometrical definition of the safety index is often used for

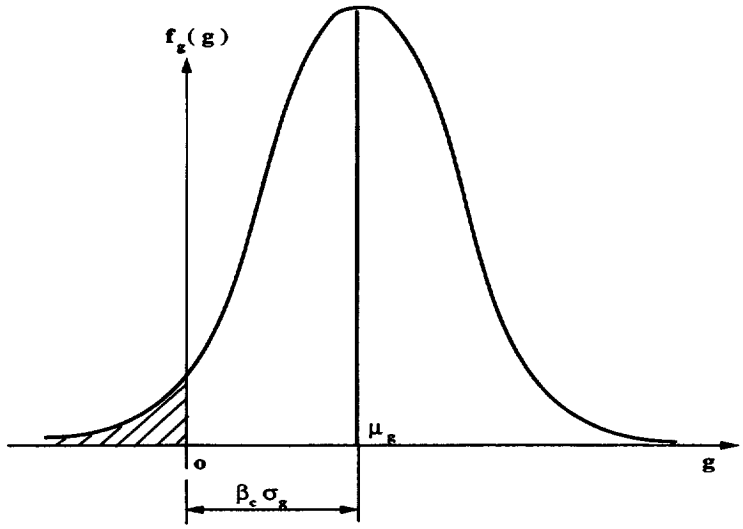


Fig. 3.3 Geometrical Illustration of the Cornell Reliability Index, β_c, (II)

describing the Cornell reliability index. A more general definition of the safety index can be given as follows. Introduce standard normalized random variables of R and S,

$$\hat{R} = \frac{R - \mu_R}{\sigma_R}, \qquad \hat{S} = \frac{S - \mu_S}{\sigma_S} \tag{3.2.11}$$

where μ_R and μ_S are the mean values of random variables R and S, respectively; and σ_R and σ_S are the standard deviations of R and S, respectively. The limit state surface $g(R,S) = R - S = 0$ in the (R,S) coordinate system can be transformed into a straight line in the standard normalized $(\hat{R}, \hat{S})$ coordinate system by substituting Eq.(3.2.11)

$$g(\hat{R}, \hat{S}) = \hat{R}\sigma_R - \hat{S}\sigma_S + (\mu_R - \mu_S) = 0 \tag{3.2.12}$$

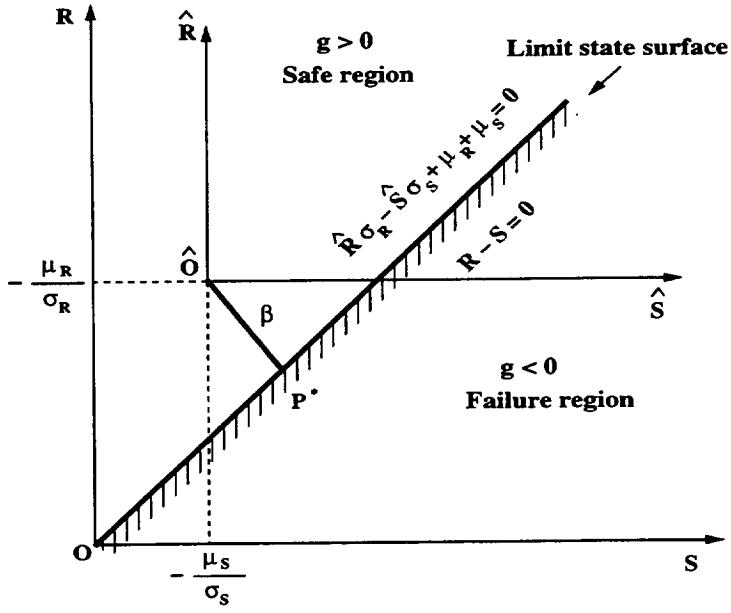


Fig. 3.4 Geometrical Illustration of Safety Index, β , (III)

The shortest distance from the origin in the $(\hat{R}, \hat{S})$ coordinate system to the failure surface $g(\hat{R}, \hat{S}) = 0$ is equal to

$$\hat{O}P^* = \frac{\mu_R - \mu_S}{\sqrt{\sigma_R^2 + \sigma_S^2}} \tag{3.2.13}$$

By comparing Eq. (3.2.10) with this equation, it is obvious that the shortest distance $\hat{O}P^*$ is the safety index β, *i.e.*,

$$\beta = \hat{O}P^* = \frac{\mu_R - \mu_S}{\sqrt{\sigma_R^2 + \sigma_S^2}} \tag{3.2.14}$$

The geometrical illustration of the safety index is given in Fig. 3.4. Therefore, a general geometric definition of the safety index β is the shortest distance from the origin to the limit state surface. This definition can be used for not only the Cornell reliability index, but also general cases, such as the following Hasofer-Lind reliability index. It is necessary to point out that even though the definition of the safety index can be used for general cases, but Eq. (3.2.14) can only be applied for the problem having the limit state function given in Eq. (3.1.4) and the normally distributed random variables R and S. For the nonlinear limit state function and non-normal distribution variables, the safety index calculations will be given later.

3.3 First-Order Second-Moment (FOSM) Method

In the above sections, the limit state function was considered simply linear and to contain just two normally distributed variables. However, in practice, the limit state function can be nonlinear, the size of the random variables X can be very large, and their distributions can be of other types such as lognormal, Weibull, extreme value, *etc.* It is necessary that all uncertainties must be contained in the joint probability density function $f_X(X)$ in calculating the failure probability and the limit state functions. In fact, it is usually extremely difficult or even impossible to construct the joint density function and/or determine each of the individual density functions because of the scarcity of statistical data. Even in the case where statistical information may be sufficient to determine these functions, it is often impractical to perform numerically the multidimensional integration over the failure region Ω by using numerical integration or Monte Carlo simulation due to the complexity, nonlinearity and time-consuming analyses of the limit state functions.

The first-order second-moment (FOSM) method simplifies the functional relationship and alleviates the above difficulties. The name first-order comes from the first-order expansion of

the function. In principle, the random variables are characterized by their mean values (first moments) and standard deviations (second moments). This is possible for any continuous mathematical form of the limit state equation. The different ways of simplifying the limit state function form different reliability analysis algorithms. As implied, inputs and outputs are expressed as the mean and standard deviation. Higher moments, which might describe skew and flatness of the distribution, are ignored.

3.3.1 Mean Value Method

In the mean value method, the limit state function is linearized by means of the Taylor series expansion (truncated at the linear terms only) at the mean value point. Assuming that the X variables are statistically independent, the first-order Taylor series expansion of the limit state function at the mean $\mu = \{\mu_{x_1}, \mu_{x_2}, ..., \mu_{x_n}\}^T$ is

$$\tilde{g}(X) \approx g(\mu) + (X - \mu)g'(\mu) \tag{3.3.1}$$

The mean value of the approximate limit state function $\tilde{g}(X)$ is

$$\begin{aligned} \mu_{\tilde{g}} &\approx E[g(\mu)] \\ &= g(\mu) \\ &= g(\mu_{x_1}, \mu_{x_2}, ..., \mu_{x_n}) \end{aligned} \tag{3.3.2}$$

The variance of the approximate limit state function $\tilde{g}(X)$ is

$$Var[\tilde{g}(X)] \approx Var[g(\mu)] + Var[(X - \mu)g'(\mu)] \tag{3.3.3}$$

Because

$$Var[g(\mu)] = 0, \qquad Var[g'(\mu)] = 0 \tag{3.3.4}$$

$$Var[(X - \mu)g'(\mu)] = Var[Xg'(\mu)] - \mu Var[g'(\mu)]$$

$$= Var[Xg'(\mu)]$$
$$= [g'(\mu)]^2 Var(X) \tag{3.3.5}$$

Therefore, the standard deviation of the approximate limit state function is

$$\begin{aligned} \sigma_{\tilde{g}(X)} &= \sqrt{Var[\tilde{g}(X)]} \\ &= \sqrt{[g'(\mu)]^2 Var(X)} \\ &= [(\sum_{i=1}^{n} \frac{\partial g}{\partial x_i}|_{X=\mu})^2 \sigma_{x_i}^2]^{1/2} \end{aligned} \tag{3.3.6}$$

The reliability index β is computed as:

$$\beta = \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} \tag{3.3.7}$$

Eq. (3.3.7) is the same as Eq. (3.2.10) if the limit state function is linear. If the limit state function is nonlinear, the approximate limit state surface is obtained by linearizing the original limit state function at the mean value point. Therefore, this method is called the mean value method, and the β given in Eq. (3.3.7) is called a mean value first-order second-moment (MVFOSM) reliability index.

In the general case with the independent variables of n-dimensional space, the failure surface is a hyperplane and can be defined as a linear failure function:

$$\tilde{g}(X) = c_0 + \sum_{i=1}^{n} c_i x_i \tag{3.3.8}$$

The reliability index given in Eq. (3.3.7) can still be used for this n-dimensional case, in which

$$\mu_{\tilde{g}} = c_0 + c_1 \mu_{x_1} + c_2 \mu_{x_2} + ... + c_n \mu_{x_n} \tag{3.3.9}$$

$$\sigma_{\tilde{g}} = \sqrt{\sum_{i=1}^{n} c_i^2 \sigma_{x_i}^2} \tag{3.3.10}$$

Example 3.1

The performance function is

$$g(x_1, x_2) = x_1^3 + x_2^3 - 18$$

in which x_1 and x_2 are the random variables with normal distributions (mean $\mu_{x_1} = \mu_{x_2} = 10.0$, standard deviation $\sigma_{x_1} = \sigma_{x_2} = 5.0$). The mean value method is used to solve the safety index β.

The mean of the linearized performance function is

$$\mu_{\tilde{g}} = g(\mu_{x_1}, \mu_{x_2}) = 1982.00$$

The standard deviation of the linearized performance function is

$$\begin{aligned} \sigma_{\tilde{g}} &= \sqrt{(\frac{\partial g(\mu_{x_1}, \mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1}, \mu_{x_2})}{\partial x_2}\sigma_{x_2})^2} \\ &= \sqrt{(3 \times 10^2 \times 5.0)^2 + (3 \times 10^2 \times 5.0)^2} \\ &= 2121.32 \end{aligned}$$

From Eq. (3.3.7), the safety index β is

$$\beta = \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} = \frac{1982.00}{2121.32} = 0.9343$$

The mean value method changes the original complex probability problem into a simple one. This method directly establishes the relationship between the reliability index and the basic parameters (means and standard deviations) of the random variables. However, there are two serious drawbacks in the mean-value FOSM method:

1. Evaluation of reliability by linearizing the limit state function about the mean values leads to erroneous estimates for performance functions with high nonlinearity, or for large coefficients of variation. This can be seen from the following mean value calculation of $\tilde{g}(X)$, assuming that the truncation of the Taylor series expansion after the first three terms is,

$$\tilde{g}(X) \approx g(\mu) + (X-\mu)g'(\mu) + \frac{(X-\mu)^2}{2}g''(\mu) \tag{3.3.11}$$

Based on Eq. (B.4), the mean value of the approximate limit state function $\tilde{g}(X)$ can be calculated as

$$\mu_{\tilde{g}} \approx E[g(\mu)] + E[(X-\mu)g'(\mu)] + E[\frac{(X-\mu)^2}{2}g''(\mu)] \tag{3.3.12}$$

Because

$$E[g(\mu)] = g(\mu) \tag{3.3.13}$$

$$\begin{aligned} E[(X-\mu)g'(\mu)] &= E[Xg'(\mu)] - E[\mu g'(\mu)] \\ &= g'(\mu)E(X) - \mu g'(\mu) \\ &= 0 \end{aligned} \tag{3.3.14}$$

$$\begin{aligned} E[\frac{(X-\mu)^2}{2}g''(\mu)] &= \frac{1}{2}g''(\mu)E[(X-\mu)^2] \\ &= \frac{1}{2}g''(\mu)Var(X) \end{aligned} \tag{3.3.15}$$

From Eq. (3.3.15), it is obvious that the second term of Eq. (3.3.11) depends on the variance of X and the second-order gradients of the limit state function. If the variance of X is small or the limit state function is close to linear, the second term of Eq. (3.3.11) can be ignored and the mean value of $\tilde{g}(X)$ is the same as Eq. (3.3.2). Otherwise, large errors in the mean value estimation will result.

2. The mean-value FOSM method fails to be invariant with different mechanically equivalent formulations of the same problem. This is a problem not only for nonlinear forms of $g(.)$, but also for certain linear forms. Example 3.2 shows that two different equivalent formulations of the limit state function for the same problem results in different safety indices.

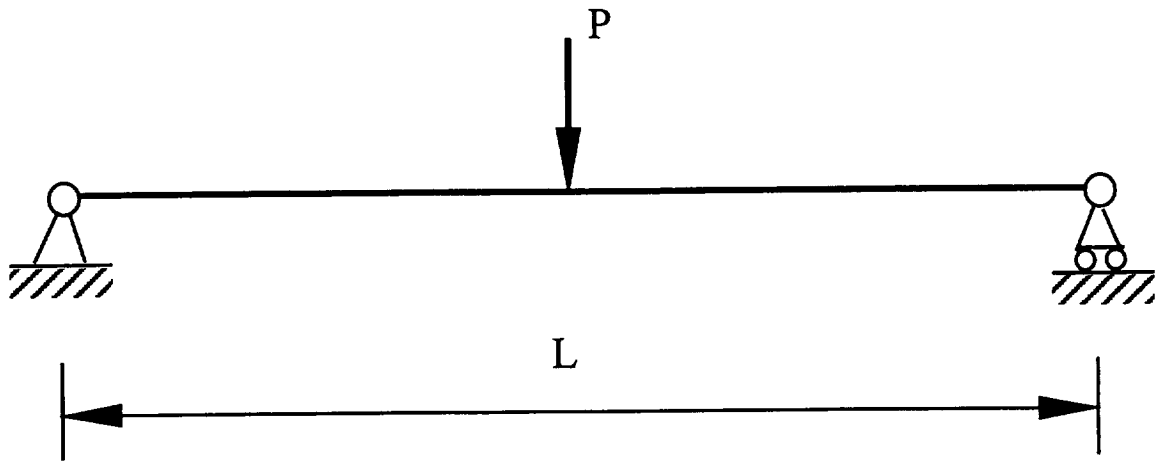


Fig. 3.5 Simply Supported Beam Loaded at the Midpoint

Example 3.2

This example is taken from Ref. [2]. Figure 3.5 shows a simply supported beam loaded at the midpoint by a concentrated force P. The length of the beam is L, and the bending moment capacity at any point along the beam is WT, where W is the plastic section modulus and T is the yield stress. All four random variables P, L, W, and T are assumed to be normal distributions. The mean values of P, L, W, and T are $10kN$, $8m$, $100 \times 10^{-6}m^3$, and $600 \times 10^3 kN/m^2$, respectively. The standard deviations of P, L, W, and T are $2kN$, $0.1m$, $2 \times 10^{-5}m^3$, and $10^5 kN/m^2$, respectively. Two different equivalent formulations of the limit state function can be given as

$$g_1(P, L, W, T) = WT - \frac{PL}{4}$$

$$g_2(P, L, W, T) = T - \frac{PL}{4W}$$

The safety index for the g_1 function is

$$\begin{aligned}\beta_1 &= \frac{\mu_{g_1}}{\sigma_{g_1}} \\ &= \frac{100 \times 10^{-6} \times 600 \times 10^3 - \frac{10\times 8}{4}}{\sqrt{(-2\times 2)^2 + (-2.5 \times 0.1)^2 + (600 \times 10^3 \times 2 \times 10^{-5})^2 + (100 \times 10^{-6} \times 10^5)^2}} \\ &= 2.48\end{aligned}$$

and the safety index for the g_2 function is

$$\begin{aligned}\beta_2 &= \frac{\mu_{g_2}}{\sigma_{g_2}} \\ &= \frac{600 \times 10^3 - \frac{10\times 8}{4\times 100\times 10^{-6}}}{\sqrt{(-2\times 10^4 \times 2)^2 + (-0.25 \times 10^2)^2 + (4 \times 10^4)^2 + (1. \times 10^5)^2}} \\ &= 3.48\end{aligned}$$

β_1 and β_2 are different even though the above two limit state equations are equivalent.

3.3.2 Advanced First-Order Second-Moment Method – Hasofer and Lind Safety Index

In the previous sections, Figs. 3.2-3.4 showed how the reliability index could be interpreted as the measure of the distance to the failure surface. In the one-dimensional case, the standard deviation of the safety margin was conveniently used as the scale. To obtain a similar scale in the case of more basic variables, Hasofer and Lind (1974) proposed a nonhomogeneous linear mapping of the set of basic variables into a set of normalized and independent variables u_i. Take the example of the stress-strength given in Section 3.1. Consider the fundamental case with the independent variables of strength, R, and stress, S, which are both normally distributed. First, Hasofer and Lind introduced the standard normalized random variables given in Eq. (3.2.11), and transformed the limit state surface $g(R,S) = R - S = 0$ in the original (R,S) coordinate system into the limit state surface given in Eq. (3.2.12) in the standard normalized $(\hat{R},\hat{S})$ coordinate system. Here, the shortest distance from the origin to the linear failure surface is

defined as the safety index β. The point $P^*(\hat{R}^*, \hat{S}^*)$ on $g(\hat{R}, \hat{S}) = 0$, which corresponds to this shortest distance, is referred to as the *checking point* or *design point.*

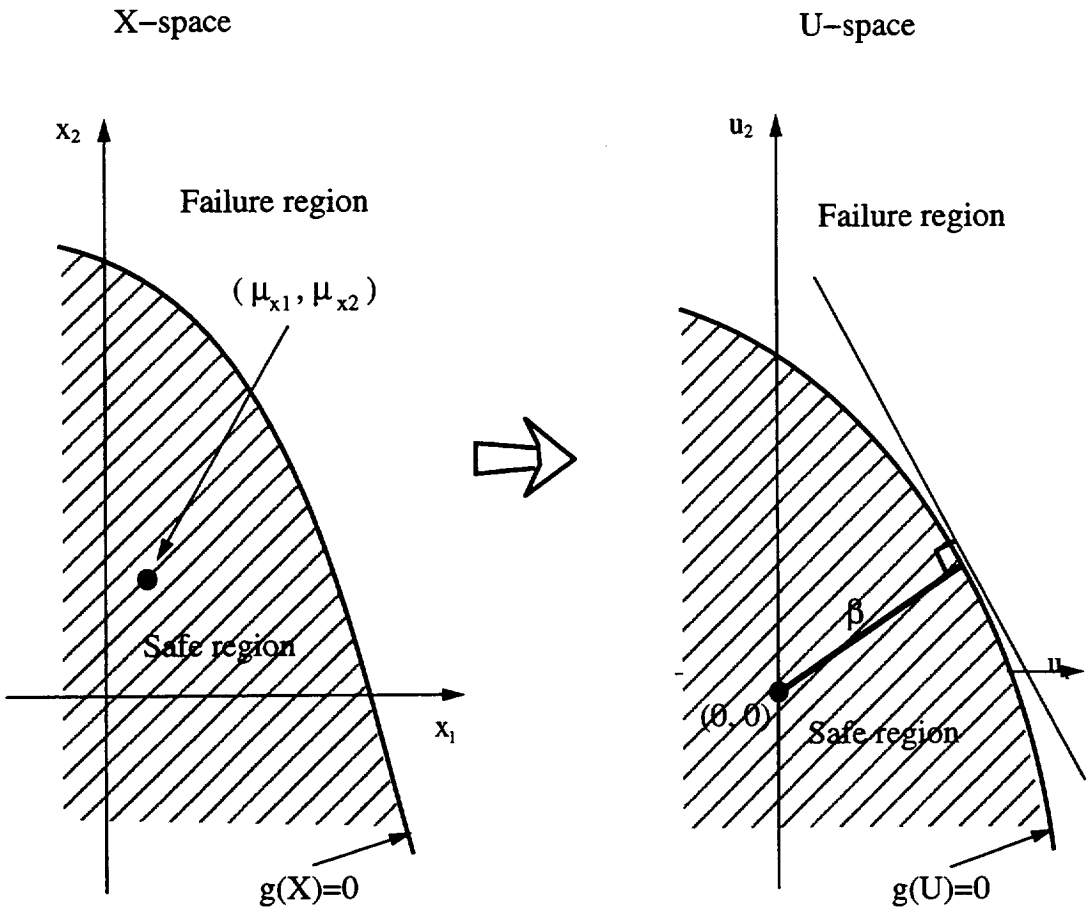


Fig. 3.6 Mapping of Failure Surface from X–space to U–space

In a general case with normally distributed and independent variables of n-dimensional space, the failure surface is a nonlinear function:

$$g(X) = g(x_1, x_2, ..., x_n) \tag{3.3.16}$$

Transform all the variables into their standardized forms:

$$u_i = \frac{x_i - \mu_{x_i}}{\sigma_{x_i}} \tag{3.3.17}$$

where μ_{x_i} and σ_{x_i} represent the mean value and standard deviation of x_i, respectively. The mean value and standard deviation of the standard normally distributed variable, u_i, are zero and unity, respectively. Any orthogonal transformation of the standard normally distributed variables, $U = \{u_1, u_2, ..., u_n\}^T$, results in a new set of normalized and uncorrelated variables. Therefore, the distributions of U are rotationally symmetric with respect to the second-moment distribution. Based on the transformation of Eq. (3.3.17), the mean value point in the original space (X-space) is mapped into the origin of the standard normal space (U-space). The failure surface $g(X) = 0$ in X-space is mapped into the corresponding failure surface $g(U) = 0$ in U-space as shown in Fig. 3.6. Due to the rotational symmetry of the second-moment representation of U, the geometrical distance from the origin in U-space to any point on $g(U) = 0$ is simply the number of standard deviations from the mean value point in X-space to the corresponding point on $g(X) = 0$. The distance to the failure surface can then be measured by the safety index function

$$\beta(U) = (U^T U)^{1/2}, \qquad U \in g(U) = 0 \tag{3.3.18}$$

The safety index β is the shortest distance from the origin to the failure surface $g(U) = 0$, i.e.

$$\beta = \min_{U \in g(U)=0} (U^T U)^{1/2} \tag{3.3.19}$$

This safety index is also called the Hasofer and Lind safety index, β_{HL}. The point $U^*(u_1^*, u_2^*, ..., u_n^*)$ on $g(U) = 0$ is a design point. The values of the safety indices given in Eqs. (3.3.7) and (3.3.19) are the same when the failure surface is a hyperplane.

The Hasofer and Lind reliability index can also be interpreted as a first-order second-moment reliability index. The value of β_{HL} is the same for the true failure surface as for the approximate

tangent hyperplane at the design. The ambiguity in the value of the first-order reliability index is thus resolved when the design point is taken as the linearization point. The resultant reliability index is a sensible measure for the distance to the failure surface.

3.3.2.1 Safety Index Model – Optimization Problem

Eq. (3.3.19) shows that the safety index β is the solution of a constrained optimization problem in the standard normal space:

$$Minimize: \qquad \beta(U) = (U^T U)^{1/2} \tag{3.3.20a}$$

$$Subject \quad to: \qquad g(U) = 0 \tag{3.3.20b}$$

There are many algorithms available that can solve this problem, such as mathematical optimization schemes or other iteration algorithms. In Ref. [3], several constrained optimization methods were used to solve this optimization problem, which included primal (feasible directions, gradient, projection, reduced gradient), penalty, dual and Lagrange methods. Each method has its advantages and disadvantages, depending upon the attributes of the method and the nature of the problem. Unlike the optimization schemes, another class of method is the iteration algorithm. In the following section, the most commonly-used iteration algorithms, the HL and HL-RF methods, are introduced to solve the reliability problems.

3.3.2.2 Solve Safety Index Model Using HL Iteration Method [4]

The HL method was proposed by Hasofer and Lind, and the HL-RF method was extended by Rackwitz and Fiessler based on the HL method to include random variable distribution information. The HL method is introduced in the following section and the HL-RF method will be given later. Assume that the limit state surface with n-dimensional normally distributed and independent random variables X is

$$g(X) = g(x_1, x_2, ..., x_n) = 0 \tag{3.3.21}$$

This limit state function can be linear or nonlinear. Based on the transformation given in Eq. (3.3.17), the limit state function given in Eq. (3.3.21) is transferred into

$$g(U) = g(\sigma_{x_1}u_1 + \mu_{x_1}, \sigma_{x_2}u_2 + \mu_{x_2}, ..., \sigma_{x_n}u_n + \mu_{x_n}) = 0 \tag{3.3.22}$$

The normal vector from the origin $\hat{O}$ to the limit state surface $g(U)$ generates an intersection point P^*, and this point has been defined as the design point before. The distance from the origin to the design point is the safety index β. The first-order Taylor series expansion of $g(U)$ at the design point $U^* = \{u_1^*, u_2^*, ..., u_n^*\}^T$ is

$$\tilde{g}(U) \approx g(U^*) + \sum_{i=1}^{n} \frac{\partial g(U^*)}{\partial u_i}(u_i - u_i^*) \tag{3.3.23}$$

From the transformation of Eq. (3.3.17), we have

$$\frac{\partial g(U)}{\partial u_i} = \frac{\partial g(X)}{\partial x_i}\sigma_{x_i} \tag{3.3.24}$$

The shortest distance from the origin to the above approximate failure surface given in Eq. (3.3.23) is

$$\hat{O}P^* = \beta = \frac{g(U^*) - \sum_{i=1}^{n} \frac{\partial g(U^*)}{\partial x_i}\sigma_{x_i}u_i^*}{\sqrt{\sum_{i=1}^{n}(\frac{\partial g(U^*)}{\partial x_i}\sigma_{x_i})^2}} \tag{3.3.25}$$

The direction cosine of the unit outward normal vector is given as

$$\begin{aligned} cos\theta_{x_i} = cos\theta_{u_i} &= -\frac{\frac{\partial g(U^*)}{\partial u_i}}{|\nabla g(U^*)|} \\ &= -\frac{\frac{\partial g(X^*)}{\partial x_i}\sigma_{x_i}}{[\sum_{i=1}^{n}(\frac{\partial g(X^*)}{\partial x_i}\sigma_{x_i})^2]^{1/2}} \\ &= \alpha_i \end{aligned} \tag{3.3.26}$$

where α_i expresses the relative effect of the corresponding random variable on the total variation. Thus, it is called the sensitivity factor. More details about α_i will be given later.

The coordinates of the point P^* are computed as

$$\begin{aligned} u_i^* &= \frac{x_i^* - \mu_{x_i}}{\sigma_{x_i}} \\ &= \hat{O}P^* cos\theta_{x_i} \\ &= \beta cos\theta_{x_i} \end{aligned} \tag{3.3.27}$$

The coordinates corresponding to P^* in the original space are

$$x_i^* = \mu_{x_i} + \beta\sigma_{x_i} cos\theta_{x_i}, \qquad (i = 1, 2, ..., n) \tag{3.3.28}$$

Since P^* is a point on the limit state surface, so

$$g(x_1^*, x_2^*, ..., x_n^*) = 0 \tag{3.3.29}$$

The main steps of the HL iteration method consist of:

1) Define the appropriate limit state function of Eq. (3.3.21);

2) Set the mean value point as an initial design point, i.e., $x_i^* = \mu_{x_i}, \quad i = 1, 2, ..., n$, and compute the gradients $\nabla g(X^*) = \{\frac{\partial g(x^*)}{\partial x_1}, \frac{\partial g(X^*)}{\partial x_2}, ..., \frac{\partial g(X^*)}{\partial x_n}\}^T$ of the limit state function at this point;

3) Compute the initial β using the mean value method (Cornell safety index), *i.e*, $\beta = \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}}$ and its direction cosine;

4) Compute a new design point X^* from Eq. (3.3.28), and function value and gradients at this new design point;

5) Compute the safety index β using Eq. (3.3.25) and the direction cosine or sensitivity factor from Eq.(3.3.26);

6) Repeat the steps (4) -(6) until the estimate of β converges;

7) Compute the coordinates of the design point or most probable failure point (MPP), X^*.

In some cases, the failure surface may contain several points corresponding to stationary values of the reliability index function. Therefore, it is necessary to use several starting points

to find all stationary values β_1, β_2, ..., β_m. This is called a multiple MPP problem. More details about the multiple MPP problem will be given later. The Hasofer and Lind safety index is

$$\beta_{HL} = min\{\beta_1, \beta_2, ..., \beta_m\} \tag{3.3.30}$$

In Eqs.(3.3.7) and (3.3.25), the difference between the mean value method and HL method is that the HL method approximates the limit state function using the first-order Taylor expansion at the design point X^* or U^* instead of the mean value point μ. Also, the mean value method doesn't require iterations, while the HL method needs several iterations to converge for nonlinear problems. Therefore, the HL method usually provides better results than the mean value method for the nonlinear problems. How well a linearized limit state function $\tilde{g}(U) = 0$ approximates a nonlinear function $g(U)$ in terms of the failure probability P_f depends on the shape of $g(U) = 0$. If it is concave towards the origin, P_f is underestimated by the hyperplane approximation. Similarly, a convex function implies overestimation. However, there is no guarantee that the HL algorithm converges in all situations. Furthermore, the Hasofer and Lind method only considers normally distributed random variables, so it can not be used for the nonnormal distributed random variables. Instead, another similar iteration method, called the RF method, which will be introduced later, needs to be used.

Note: The HL method described here is a little different from the one given in Ref. [2]. In Ref. [2], the coordinates of the design point X^* are determined first, and the safety index β is calculated by using Eq. (3.3.25). The iteration process stops when the estimation of x_i^* is stable.

Example 3.3a

In this example, the performance function, mean values, standard deviations, and distributions of both random variables are the same as in Example 3.1. The HL method is used to solve the safety index β.

(1) Iteration 1:

(a) Set the mean value point as an initial design point and required β convergence tolerance as $\varepsilon_r = 0.001$. Compute the limit state function value and gradients at the mean value point

$$\begin{aligned} g(X^*) &= g(\mu_{x_1}, \mu_{x_2}) = \mu_{x_1}^3 + \mu_{x_2}^3 - 18 \\ &= 10.0^3 + 10.0^3 - 18 \\ &= 1982.00 \end{aligned}$$

$$\frac{\partial g}{\partial x_1}|_\mu = 3\mu_{x_1}^2 = 3 \times 10^2 = 300$$
$$\frac{\partial g}{\partial x_2}|_\mu = 3\mu_{x_2}^2 = 3 \times 10^2 = 300$$

(b) Compute the initial β using the mean value method and its direction cosine α_i

$$\begin{aligned} \beta_1 &= \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} \\ &= \frac{g(X^*)}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= \frac{1982.00}{\sqrt{(300 \times 5.0)^2 + (300 \times 5.0)^2}} \\ &= 0.9343 \end{aligned}$$

$$\begin{aligned} \alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_\mu \sigma_{x_1}}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= -\frac{300 \times 5.0}{\sqrt{(300 \times 5.0)^2 + (300 \times 5.0)^2}} \\ &= -0.7071 \end{aligned}$$

$$\begin{aligned}\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_{\mu}\sigma_{x_2}}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2+(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= -\frac{300\times 5.0}{\sqrt{(300\times 5.0)^2+(300\times 5.0)^2}} \\ &= -0.7071\end{aligned}$$

(c) Compute a new design point X^* from Eq. (3.3.28)

$$\begin{aligned}x_1^* &= \mu_{x_1}+\beta_1\sigma_{x_1}\alpha_1 \\ &= 10.0+0.9343\times 5.0\times(-0.7071) \\ &= 6.6967\end{aligned}$$

$$\begin{aligned}x_2^* &= \mu_{x_2}+\beta_1\sigma_{x_2}\alpha_2 \\ &= 10.0+0.9343\times 5.0\times(-0.7071) \\ &= 6.6967\end{aligned}$$

$$u_1^* = \frac{x_1^*-\mu_{x_1}}{\sigma_{x_1}} = \frac{6.6967-10.0}{5.0} = -0.6607$$
$$u_2^* = \frac{x_2^*-\mu_{x_2}}{\sigma_{x_2}} = \frac{6.6967-10.0}{5.0} = -0.6607$$

(2) Iteration 2:

(a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3+(x_2^*)^3-18 = 6.6967^3+6.6967^3-18 = 582.63$$
$$\frac{\partial g}{\partial x_1}|_{X^*} = 3\times(x_1^*)^2 = 3\times 6.6967^2 = 134.5374$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 6.6967^2 = 134.5374$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_2 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{582.63 - 134.5374 \times 5.0 \times (-0.6607) - 134.5374 \times 5.0 \times (-0.6607)}{\sqrt{(134.5347 \times 5.0)^2 + (134.5347 \times 5.0)^2}} \\
&= 1.5468
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{134.5374 \times 5.0}{\sqrt{(134.5374 \times 5.0)^2 + (134.5374 \times 5.0)^2}} \\
&= -0.7071
\end{aligned}$$

$$\alpha_2 = \alpha_1 = -0.7071$$

(c) Compute a new design point X^*

$$\begin{aligned}
x_1^* &= \mu_{x_1} + \beta_2 \sigma_{x_1} \alpha_1 \\
&= 10.0 + 1.5468 \times 5.0 \times (-0.7071) \\
&= 4.5313
\end{aligned}$$

$$x_2^* = x_1^* = 4.5313$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{4.5313 - 10.0}{5.0} = -1.0937$$

$$u_2^* = u_1^* = -1.0937$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_2 - \beta_1|}{\beta_1} = \frac{1.5468 - 0.9343}{0.9343} = 0.6556$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(3) Iteration 3:

a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 4.5313^3 + 4.5313^3 - 18 = 168.08$$

$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 4.5313^2 = 61.598$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 4.5313^2 = 61.598$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_3 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{168.08 - 61.598 \times 5.0 \times (-1.0937) - 61.598 \times 5.0 \times (-1.0937)}{\sqrt{(61.598 \times 5.0)^2 + (61.598 \times 5.0)^2}} \\
&= 1.9327
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{61.598 \times 5.0}{\sqrt{(61.598 \times 5.0)^2 + (61.598 \times 5.0)^2}} \\
&= -0.7071
\end{aligned}$$

$$\alpha_2 = \alpha_1 = -0.7071$$

(c) Compute a new design point X^*

$$\begin{aligned} x_1^* &= \mu_{x_1} + \beta_3 \sigma_{x_1} \alpha_1 \\ &= 10.0 + 1.9327 \times 5.0 \times (-0.7071) \\ &= 3.1670 \end{aligned}$$

$$x_2^* = x_1^* = 3.1670$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{3.1670 - 10.0}{5.0} = -1.3666$$

$$u_2^* = u_1^* = -1.3666$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_3 - \beta_2|}{\beta_2} = \frac{1.9327 - 1.5468}{1.5468} = 0.2495$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(4) Iteration 4:

a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 3.1670^3 + 3.1670^3 - 18 = 45.529$$

$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 3.1670^2 = 30.0897$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 3.1670^2 = 30.0897$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}\beta_4 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i}\sigma_{x_i}u_i^*}{\sqrt{\sum_{i=1}^{2}(\frac{\partial g(X^*)}{\partial x_i}\sigma_{x_i})^2}} \\ &= \frac{45.529 - 30.0897 \times 5.0 \times (-1.3666) - 30.0897 \times 5.0 \times (-1.3666)}{\sqrt{(30.0897 \times 5.0)^2 + (30.0897 \times 5.0)^2}} \\ &= 2.1467\end{aligned}$$

$$\begin{aligned}\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*}\sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*}\sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*}\sigma_{x_2})^2}} \\ &= -\frac{30.0897 \times 5.0}{\sqrt{(30.0897 \times 5.0)^2 + (30.0897 \times 5.0)^2}} \\ &= -0.7071\end{aligned}$$

$$\alpha_2 = \alpha_1 = -0.7071$$

(c) Compute a new design point X^*

$$\begin{aligned}x_1^* &= \mu_{x_1} + \beta_4\sigma_{x_1}\alpha_1 \\ &= 10.0 + 2.1467 \times 5.0 \times (-0.7071) \\ &= 2.4104\end{aligned}$$

$$x_2^* = x_1^* = 2.4104$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{2.4104 - 10.0}{5.0} = -1.5179$$

$$u_2^* = u_1^* = -1.5179$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_4 - \beta_3|}{\beta_3} = \frac{2.1467 - 1.9327}{1.9327} = 0.1107$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(5) Iteration 5:

a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 2.4104^3 + 2.4104^3 - 18 = 10.01$$

$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 2.4104^2 = 17.43$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 2.4104^2 = 17.43$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_5 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{10.01 - 17.43 \times 5.0 \times (-1.5179) - 17.43 \times 5.0 \times (-1.5179)}{\sqrt{(17.43 \times 5.0)^2 + (17.43 \times 5.0)^2}} \\
&= 2.2279
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{17.43 \times 5.0}{\sqrt{(17.43 \times 5.0)^2 + (17.43 \times 5.0)^2}} \\
&= -0.7071
\end{aligned}$$

$$\alpha_2 = \alpha_1 = -0.7071$$

(c) Compute a new design point X^*

$$\begin{aligned} x_1^* &= \mu_{x_1} + \beta_4 \sigma_{x_1} \alpha_1 \\ &= 10.0 + 2.2279 \times 5.0 \times (-0.7071) \\ &= 2.1233 \end{aligned}$$

$$x_2^* = x_1^* = 2.1233$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{2.1233 - 10.0}{5.0} = -1.5753$$

$$u_2^* = u_1^* = -1.5753$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_5 - \beta_4|}{\beta_4} = \frac{2.2279 - 2.1467}{2.1467} = 0.036$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(6) Iteration 6:

(a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 2.1233^3 + 2.1233^3 - 18 = 1.1451$$

$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 2.1451^2 = 13.8044$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 2.1451^2 = 13.8044$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_5 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{1.1451 - 13.8044 \times 5.0 \times (-1.5753) - 13.8044 \times 5.0 \times (-1.5753)}{\sqrt{(13.8044 \times 5.0)^2 + (13.8044 \times 5.0)^2}} \\
&= 2.2398
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{13.8044 \times 5.0}{\sqrt{(13.8044 \times 5.0)^2 + (13.8044 \times 5.0)^2}} \\
&= -0.7071
\end{aligned}$$

$$\alpha_2 = \alpha_1 = -0.7071$$

(c) Compute a new design point X^*

$$\begin{aligned}
x_1^* &= \mu_{x_1} + \beta_5 \sigma_{x_1} \alpha_1 \\
&= 10.0 + 2.2398 \times 5.0 \times (-0.7071) \\
&= 2.0810
\end{aligned}$$

$$x_2^* = x_1^* = 2.0810$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{2.0810 - 10.0}{5.0} = -1.5838$$

$$u_2^* = u_1^* = -1.5838$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_6 - \beta_5|}{\beta_5} = \frac{2.2398 - 2.2279}{2.2279} = 0.005$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(7) Iteration 7:

a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 2.0810^3 + 2.0810^3 - 18 = 0.023$$

$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 2.0810^2 = 12.9917$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 2.0810^2 = 12.9917$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_5 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{0.023 - 12.9917 \times 5.0 \times (-1.5838) - 12.9917 \times 5.0 \times (-1.5838)}{\sqrt{(12.9917 \times 5.0)^2 + (12.9917 \times 5.0)^2}} \\
&= 2.2401
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{12.9917 \times 5.0}{\sqrt{(12.9917 \times 5.0)^2 + (12.9917 \times 5.0)^2}} \\
&= -0.7071
\end{aligned}$$

$$\alpha_2 = \alpha_1 = -0.7071$$

(c) Compute a new design point X^*

$$\begin{aligned} x_1^* &= \mu_{x_1} + \beta_6 \sigma_{x_1} \alpha_1 \\ &= 10.0 + 2.2401 \times 5.0 \times (-0.7071) \\ &= 2.0801 \end{aligned}$$

$$x_2^* = x_1^* = 2.0801$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{2.0801 - 10.0}{5.0} = -1.5840$$

$$u_2^* = u_1^* = -1.5840$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_7 - \beta_6|}{\beta_6} = \frac{2.2401 - 2.2398}{2.22398} = 0.0001$$

Since $\varepsilon < \varepsilon_r$, stop the process.

The safety index β is 2.2401. Since the limit state function value at the MPP X^* is close to zero, this safety index can be considered as the shortest distance from the origin to the limit state surface. Compared to the safety index $\beta = 0.9343$ obtained from the Mean Value method given in Example 3.1, the safety index computed from the HL method is much more accurate for this highly nonlinear problem.

Example 3.3b

In this example, the performance function, the mean value of x_1, the standard deviations, and the distributions of both random variables are the same as in Example 3.1. The only difference between Examples 3.3a and 3.3b is that the mean value of x_2 is 9.9 instead of 10.0. The HL method is used to solve the safety index β.

(1) Iteration 1:

(a) Set the mean value point as an initial design point and required β convergence tolerance as $\varepsilon_r = 0.001$. Compute the limit state function value and gradients at the mean value point

$$\begin{aligned} g(X^*) &= g(\mu_{x_1}, \mu_{x_2}) = \mu_{x_1}^3 + \mu_{x_2}^3 - 18 \\ &= 10.0^3 + 9.9^3 - 18 \\ &= 1952.299 \end{aligned}$$

$$\frac{\partial g}{\partial x_1}|_\mu = 3\mu_{x_1}^2 = 3 \times 10^2 = 300$$
$$\frac{\partial g}{\partial x_2}|_\mu = 3\mu_{x_2}^2 = 3 \times 9.9^2 = 294.03$$

(b) Compute the initial β using the mean value method and its direction cosine α_i

$$\begin{aligned} \beta_1 &= \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} \\ &= \frac{g(X^*)}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= \frac{1952.299}{\sqrt{(300 \times 5.0)^2 + (294.03 \times 5.0)^2}} \\ &= 0.9295 \end{aligned}$$

$$\begin{aligned} \alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_\mu \sigma_{x_1}}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= -\frac{300 \times 5.0}{\sqrt{(300 \times 5.0)^2 + (294.03 \times 5.0)^2}} \\ &= -0.7142 \end{aligned}$$

$$\begin{aligned}\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_{\mu}\sigma_{x_2}}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2+(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= -\frac{294.03\times 5.0}{\sqrt{(300\times 5.0)^2+(294.03\times 5.0)^2}} \\ &= -0.7000\end{aligned}$$

c) Compute a new design point X^* from Eq. (3.3.28)

$$\begin{aligned}x_1^* &= \mu_{x_1}+\beta_1\sigma_{x_1}\alpha_1 \\ &= 10.0+0.9295\times 5.0\times(-0.7142) \\ &= 6.6808\end{aligned}$$

$$\begin{aligned}x_2^* &= \mu_{x_2}+\beta_1\sigma_{x_2}\alpha_2 \\ &= 9.9+0.9295\times 5.0\times(-0.7000) \\ &= 6.6468\end{aligned}$$

$$u_1^* = \frac{x_1^*-\mu_{x_1}}{\sigma_{x_1}} = \frac{6.6808-10.0}{5.0} = -0.6638$$
$$u_2^* = \frac{x_2^*-\mu_{x_2}}{\sigma_{x_2}} = \frac{6.6468-9.9}{5.0} = -0.6506$$

(2) Iteration 2:

(a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3+(x_2^*)^3-18 = 6.6808^3+6.6468^3-18 = 573.8398$$
$$\frac{\partial g}{\partial x_1}|_{X^*} = 3\times(x_1^*)^2 = 3\times 6.6808^2 = 133.8982$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 6.6468^2 = 132.5409$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_2 &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{573.8398 - 133.8982 \times 5.0 \times (-0.6638) - 132.5409 \times 5.0 \times (-0.6506)}{\sqrt{(133.8982 \times 5.0)^2 + (132.5409 \times 5.0)^2}} \\
&= 1.5387
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{133.8982 \times 5.0}{\sqrt{(133.8982 \times 5.0)^2 + (132.5409 \times 5.0)^2}} \\
&= -0.7107
\end{aligned}$$

$$\begin{aligned}
\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{132.5409 \times 5.0}{\sqrt{(133.8982 \times 5.0)^2 + (132.5409 \times 5.0)^2}} \\
&= -0.7035
\end{aligned}$$

(c) Compute a new design point X^*

$$\begin{aligned}
x_1^* &= \mu_{x_1} + \beta_2 \sigma_{x_1} \alpha_1 \\
&= 10.0 + 1.5387 \times 5.0 \times (-0.7107) \\
&= 4.5323
\end{aligned}$$

$$
\begin{aligned}
x_2^* &= \mu_{x_2} + \beta_2 \sigma_{x_2} \alpha_2 \\
&= 9.9 + 1.5387 \times 5.0 \times (-0.7035) \\
&= 4.4877
\end{aligned}
$$

$$
u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{4.5323 - 10.0}{5.0} = -1.0935
$$
$$
u_2^* = \frac{x_2^* - \mu_{x_2}}{\sigma_{x_2}} = \frac{4.4877 - 9.9}{5.0} = -1.0825
$$

(d) Check β convergence

$$
\varepsilon = \frac{|\beta_2 - \beta_1|}{\beta_1} = \frac{1.5387 - 0.9295}{0.9295} = 0.6554
$$

Since $\varepsilon > \varepsilon_r$, continue the process.

.....

(21) Iteration 21:

(a) Compute the limit state function value and its gradients at X^*

$$
g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 8.5253^3 + 4.2597^3 - 18 = 678.9088
$$
$$
\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 8.5253^2 = 218.0401
$$
$$
\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 4.2597^2 = 54.4352
$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_{21} &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{678.9088 - 218.0401 \times 5.0 \times (-0.2949) - 54.4352 \times 5.0 \times (-1.1281)}{\sqrt{(218.0401 \times 5.0)^2 + (54.4352 \times 5.0)^2}} \\
&= 1.1636
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{218.0401 \times 5.0}{\sqrt{(218.0401 \times 5.0)^2 + (54.4352 \times 5.0)^2}} \\
&= -0.9702
\end{aligned}$$

$$\begin{aligned}
\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{54.4352 \times 5.0}{\sqrt{(218.0401 \times 5.0)^2 + (54.4352 \times 5.0)^2}} \\
&= -0.2422
\end{aligned}$$

(c) Compute a new design point X^*

$$\begin{aligned}
x_1^* &= \mu_{x_1} + \beta_{21} \sigma_{x_1} \alpha_1 \\
&= 10.0 + 1.1636 \times 5.0 \times (-0.9702) \\
&= 4.3553
\end{aligned}$$

$$\begin{aligned}
x_2^* &= \mu_{x_2} + \beta_{21} \sigma_{x_2} \alpha_2 \\
&= 9.9 + 1.1636 \times 5.0 \times (-0.2422) \\
&= 8.4908
\end{aligned}$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{4.3553 - 10.0}{5.0} = -1.1289$$
$$u_2^* = \frac{x_2^* - \mu_{x_2}}{\sigma_{x_2}} = \frac{8.4908 - 9.9}{5.0} = -0.2818$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_{21} - \beta_{20}|}{\beta_{20}} = \frac{|1.1636 - 1.1660|}{1.1660} = 0.002$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(22) Iteration 22:

(a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 4.3553^3 + 8.4908^3 - 18 = 676.7346$$
$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 4.3553^2 = 56.9049$$
$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 8.4908^2 = 216.2786$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_{22} &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{676.7346 - 56.9049 \times 5.0 \times (-1.1289) - 216.2786 \times 5.0 \times (-0.2818)}{\sqrt{(56.9049 \times 5.0)^2 + (216.2786 \times 5.0)^2}} \\
&= 1.1650
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{56.9049 \times 5.0}{\sqrt{(56.9049 \times 5.0)^2 + (216.2786 \times 5.0)^2}} \\
&= -0.2544
\end{aligned}$$

$$\begin{aligned}
\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_{X^*}\sigma_{x_2}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*}\sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*}\sigma_{x_2})^2}} \\
&= -\frac{216.2786 \times 5.0}{\sqrt{(56.9049 \times 5.0)^2 + (216.2786 \times 5.0)^2}} \\
&= -0.9671
\end{aligned}$$

(c) Compute a new design point X^*

$$\begin{aligned}
x_1^* &= \mu_{x_1} + \beta_{21}\sigma_{x_1}\alpha_1 \\
&= 10.0 + 1.1650 \times 5.0 \times (-0.2544) \\
&= 8.5178
\end{aligned}$$

$$\begin{aligned}
x_2^* &= \mu_{x_2} + \beta_{21}\sigma_{x_2}\alpha_2 \\
&= 9.9 + 1.1650 \times 5.0 \times (-0.9671) \\
&= 4.2666
\end{aligned}$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{8.5178 - 10.0}{5.0} = -0.2964$$
$$u_2^* = \frac{x_2^* - \mu_{x_2}}{\sigma_{x_2}} = \frac{4.2666 - 9.9}{5.0} = -1.1267$$

(d) Check β convergence

$$\varepsilon = \frac{|\beta_{22} - \beta_{21}|}{\beta_{21}} = \frac{1.1650 - 1.1636}{1.1636} = 0.0012$$

Since $\varepsilon > \varepsilon_r$, continue the process.

(23) Iteration 23:

(a) Compute the limit state function value and its gradients at X^*

$$g(X^*) = (x_1^*)^3 + (x_2^*)^3 - 18 = 8.5178^3 + 4.2666^3 - 18 = 677.655$$

$$\frac{\partial g}{\partial x_1}|_{X^*} = 3 \times (x_1^*)^2 = 3 \times 8.5178^2 = 217.6582$$

$$\frac{\partial g}{\partial x_2}|_{X^*} = 3 \times (x_2^*)^2 = 3 \times 4.2666^2 = 54.61056$$

(b) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_{23} &= \frac{g(X^*) - \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i} u_i^*}{\sqrt{\sum_{i=1}^{2} (\frac{\partial g(X^*)}{\partial x_i} \sigma_{x_i})^2}} \\
&= \frac{677.655 - 217.6582 \times 5.0 \times (-0.2964) - 54.61056 \times 5.0 \times (-1.1267)}{\sqrt{(217.6582 \times 5.0)^2 + (54.61056 \times 5.0)^2}} \\
&= 1.1657
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{217.6582 \times 5.0}{\sqrt{(217.6582 \times 5.0)^2 + (54.61056 \times 5.0)^2}} \\
&= -0.9699
\end{aligned}$$

$$\begin{aligned}
\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2}}{\sqrt{(\frac{\partial g}{\partial x_1}|_{X^*} \sigma_{x_1})^2 + (\frac{\partial g}{\partial x_2}|_{X^*} \sigma_{x_2})^2}} \\
&= -\frac{54.61056 \times 5.0}{\sqrt{(217.6582 \times 5.0)^2 + (54.61056 \times 5.0)^2}} \\
&= -0.2434
\end{aligned}$$

(c) Compute a new design point X^*

$$
\begin{aligned}
x_1^* &= \mu_{x_1} + \beta_{21}\sigma_{x_1}\alpha_1 \\
&= 10.0 + 1.1657 \times 5.0 \times (-0.9699) \\
&= 4.3468
\end{aligned}
$$

$$
\begin{aligned}
x_2^* &= \mu_{x_2} + \beta_{21}\sigma_{x_2}\alpha_2 \\
&= 9.9 + 1.1657 \times 5.0 \times (-0.2434) \\
&= 8.4816
\end{aligned}
$$

$$
u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{4.3468 - 10.0}{5.0} = -1.1306
$$

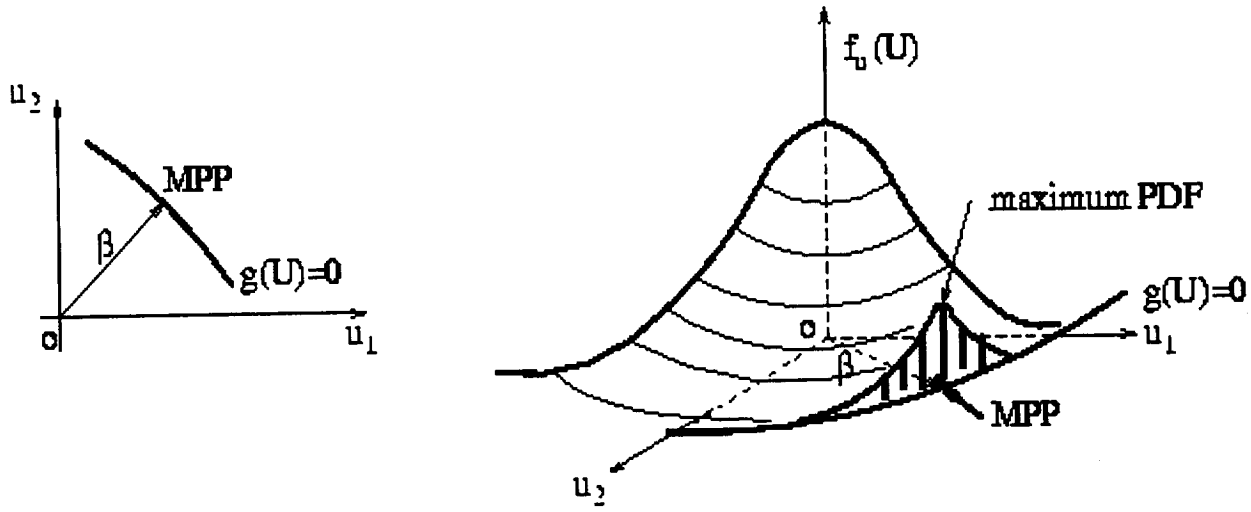
$$
u_2^* = \frac{x_2^* - \mu_{x_2}}{\sigma_{x_2}} = \frac{8.4816 - 9.9}{5.0} = -0.2837
$$

(d) Check β convergence

$$
\varepsilon = \frac{|\beta_{23} - \beta_{22}|}{\beta_{22}} = \frac{1.1657 - 1.1650}{1.1650} = 0.0006
$$

Since $\varepsilon < \varepsilon_r$, stop the process.

The safety index converges after 23 iterations; however, the MPP is not on the limit state surface ($g(X^*) = 677.655$). Also, from iterations 21, 22 and 23, the design point X^* oscillates. If a convergence check for determining whether the MPP is on the surface or not is added, the process will continue. However, no final MPP on the surface can be found after hundreds of iterations due to the oscillation. From this example, it is clear that the HL method may not converge in some cases due to its linear approximation. A more efficient method can be used to deal with this problem, and the correct safety index for this example is given in Example 3.8.



(a) Two–dimension **(b) Three–dimension**

Fig. 3.7 Most Probable failure Point (MPP)

3.3.2.3 Most Probable failure Point (MPP)

As mentioned in Section 3.3.2, the point $U^*(u_1^*, u_2^*, ..., u_n^*)$ corresponding to the shortest distance from the origin to the failure surface $g(U) = 0$ is defined as the design point. Because of rotational symmetry and the Hasofer-Lind transformation, the design point in U-space represents the point of greatest probability density or maximum likelihood (Fig. 3.7). It makes the most significant contribution to the nominal failure probability $P_f = \Phi(-\beta)$, so this design point is also called the most probable failure point (MPP).

The MPP is important in structural analyses. For the problems having a single stationary point, as shown in Fig. 3.7, the MPP corresponds to the greatest probability density in the failure region and has the maximum failure probability. That is, it implies the most possible failure design in the structures due to the uncertainties. Searching the MPP on the limit state surface is a key step in the HL method. The improvement of the HL method compared with

the MVFOSM also comes from changing the expanding point from the mean value point to the MPP.

3.3.2.4 Sensitivity Factors

As mentioned in section 3.3.2.2, the direction cosine of the unit outward normal vector of the limit state function α_i given in Eq. (3.3.26) is defined as the sensitivity factor. The sensitivity calculation of the failure probability or the safety index to small changes in random variables usually gives useful information for studying the statistical variation of the response. The sensitivity factor shows the relative importance of each random variable to the structural failure probability.

In Eq. (3.3.26), the physical meaning of α_i implies the relative contribution of each random variable to the failure probability (Fig. 3.8). For example, the larger the α_i value is, the higher the contribution towards the failure probability due to

$$\alpha_1^2 + \alpha_2^2 + ... + \alpha_n^2 = 1 \tag{3.3.31}$$

In fact, α_i is the sensitivity of the safety index β at MPP. From the definition of β as the distance from the origin to the limit state surface $g(U) = 0$, it follows that

$$\frac{\partial \beta}{\partial u_i} = \frac{\partial}{\partial u_i}\sqrt{u_1^2 + u_2^2 +, ..., +u_n^2} = \frac{u_i}{\beta} = \alpha_i$$

$$i = 1, 2, ..., n \tag{3.3.32}$$

The sensitivity factors for the failure probability P_f are

$$\frac{\partial P_f}{\partial u_i} = \frac{\partial}{\partial u_i}\Phi(-\beta) = \phi(-\beta)\frac{\partial \beta}{\partial u_i} \tag{3.3.33}$$

where $\phi(.)$ represents the standard normal density function.

In Eq.(3.3.26), $\frac{\partial g(X)}{\partial x_i}$ represents the sensitivity of the performance function $g(X)$, which measures the change in the performance function to the change in the physical random variables. However, the sensitivity of the safety index β represents the change in the safety index due to

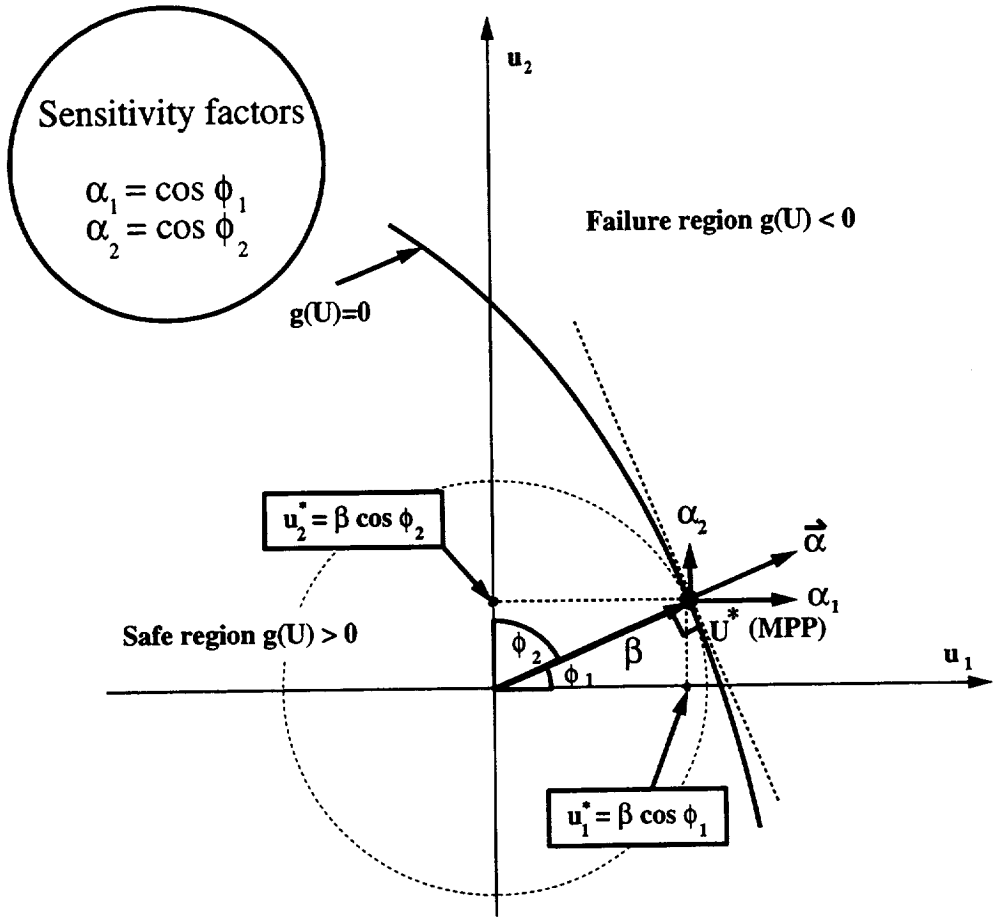


Fig. 3.8 Sensitivity Factors

the change in random variables and the uncertainty, that is, it depends on the sensitivity of the performance function $\frac{\partial g(X)}{\partial x_i}$ and the standard deviation of random variables σ_i (from Eq. 3.3.26).

In summary, what you are getting by computing the α_i is the sensitivity of the safety index with respect to u_i, which has two major functions. First, these sensitivity factors show the relative contributions of the random variables to the safety index or the failure probability. Second, the sign of the sensitivity factor gives the relation between the performance function and the physical variables. A positive α_i means that the performance function $g(U)$ decreases as the random variables increase, and a negative factor means $g(U)$ increases as the random variables increase. If you are looking for the sensitivity of the failure probability, it will have the same direction, but you need to multiply α_i with the probability density function value (Eq. 3.3.33). For example, the limit state function is

$$g(X) = \bar{S} - S(X) \tag{3.3.34}$$

where $\bar{S}$ is the allowable stress and $S(X)$ is the structural stress. If α_i is positive, it means $\frac{\partial g}{\partial x_i}$ is negative and $\frac{\partial S(X)}{\partial x_i}$ is positive. In very simplified terms, if the random variable increases, $S(X)$ increases and the failure probability increases.

3.3.2.5 Transformation of Nonnormal Variables

In the Hasofer-Lind method, the random variables X are assumed as normally distributed. Even when the limit state function $g(X)$ is linear, the structural probability calculation given in Eq. (3.2.9) is inappropriate in the nonnormal cases. However, many structural reliability problems involve nonnormal random variables. It is necessary to find a way to solve these nonnormal problems. There are many methods available for conducting the transformation, such as Rosenblatt [5], Hohenbichler and Rackwitz [6], *etc.* A simple and approximate transformation called equivalent normal distribution or the normal tail approximation is described below. The main advantages of this transformation are:

(1). It does not require the multidimensional integration;

(2). Transformation of the nonnormal variables into equivalent normal variables has been accomplished prior to the solution of Eqs.(3.3.21) - (3.3.29);

(3). Eq. (3.2.9) for calculation of the structural probability is retained;

(4). It often yields excellent agreement with the exact solution of the multi-dimensional integral of probability formula. When the variables are mutually independent, the transformation is given as

$$u_i = \Phi^{-1}[F_{x_i}(x_i)] \tag{3.3.35}$$

where $\Phi^{-1}[.]$ is the reversal of $\Phi[.]$. $F_{x_i}(x_i)$ is non-normal and $f_{x_i}(x_i)$ is normal distribution.

One way to get the equivalent normal distribution is to use the Taylor series expansion of the transformation at the MPP X^*, neglecting nonlinear terms [7],

$$u_i = \Phi^{-1}[F_{x_i}(x_i^*)] + \frac{\partial}{\partial x_i}(\Phi^{-1}[F_{x_i}(x_i)])|_{x_i^*}(x_i - x_i^*) \tag{3.3.36}$$

where

$$\frac{\partial}{\partial x_i}\Phi^{-1}[F_{x_i}(x_i)] = \frac{f_{x_i}(x_i)}{\phi(\Phi^{-1}[F_{x_i}(x_i)])} \tag{3.3.37}$$

Upon substituting (3.3.37) into (3.3.36) and rearranging,

$$u_i = \frac{x_i - [x_i^* - \Phi^{-1}[F_{x_i}(x_i^*)]\phi(\Phi^{-1}[F_{x_i}(x_i^*)])/f_{x_i}(x_i^*)]}{\phi(\Phi^{-1}[F_{x_i}(x_i^*)])/f_{x_i}(x_i^*)} \tag{3.3.38a}$$

which can be written as

$$u_i = \frac{x_i - \mu_{x_i'}}{\sigma_{x_i'}} \tag{3.3.38b}$$

where $F_{x_i}(x_i)$ is the marginal cumulative distribution function, $f_{x_i}(x_i)$ is the probability density function, and $\mu_{x_i'}$ and $\sigma_{x_i'}$ are the equivalent means and standard deviations of the approximate normal distributions. They are given as

$$\sigma_{x_i'} = \frac{\phi(\Phi^{-1}[F_{x_i}(x_i^*)])}{f_{x_i}(x_i^*)} \tag{3.3.39a}$$

$$\mu_{x_i'} = x_i^* - \Phi^{-1}[F_{x_i}(x_i^*)]\sigma_{x_i'} \tag{3.3.39b}$$

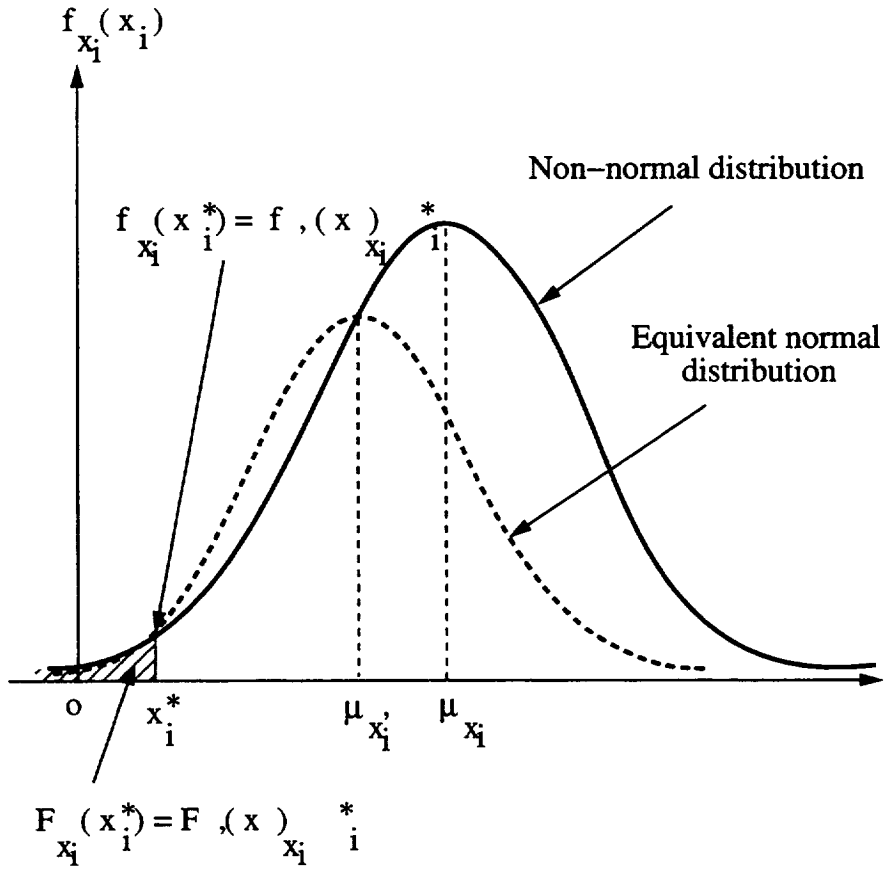


Fig. 3.9 Normal Tail Approximation

Another way to get the equivalent normal distribution is to match the cumulative distribution functions and probability density function of the original non-normal and the approximate or equivalent normal random variable distributions at the MPP [5]. Assuming that x_i' is an equivalent normally distributed random variable, the cumulative distribution function values of x_i and x_i' at x_i^* are equal, *i.e.*,

$$F_{x_i}(x_i^*) = F_{x_i'}(x_i^*) \tag{3.3.40}$$

or

$$F_{x_i}(x_i^*) = \Phi(\frac{x_i^* - \mu_{x_i'}}{\sigma_{x_i'}}) \tag{3.3.41}$$

so

$$\mu_{x_i'} = x_i^* - \Phi^{-1}[F_{x_i}(x_i^*)]\sigma_{x_i'} \tag{3.3.42}$$

The probability density function value of x_i and x_i' at x_i^* are equal, *i.e.*,

$$f_{x_i}(x_i^*) = f_{x_i'}(x_i^*) \tag{3.3.43}$$

or

$$f_{x_i}(x_i^*) = \frac{1}{\sigma_{x_i'}}\phi(\frac{x_i^* - \mu_{x_i'}}{\sigma_{x_i'}}) \tag{3.3.44}$$

From Eqs. (3.3.42) and (3.3.44), the equivalent mean $\mu_{x_i'}$ and standard deviation $\sigma_{x_i'}$ of the approximate normal distributions are derived as Eqs.(3.3.39a) and (3.3.39b). This normal tail approximation is shown in Fig. 3.9.

Using Equations (3.3.39a) and (3.3.39b), the transformation of the random variables from the X-space to the U-space can be easily performed, and the performance function $G(U)$ in U-space is approximately obtained.

Example 3.4

x is a random variable having a lognormal distribution. Its mean value and standard deviation are $\mu_x = 120$ and $\sigma_x = 12$, respectively. Calculate the mean value and standard deviation of the equivalent normal distribution variable x' at $x^* = 80.0402$.

(1) Compute the mean value, μ_y, and standard deviation, σ_y of a normally distributed variable y ($y = lnx$) using Eqs. (1.53) and (1.54)

$$\begin{aligned}\sigma_y &= \sqrt{ln[(\frac{\sigma_x}{\mu_x})^2 + 1]} \\ &= \sqrt{ln[(\frac{12}{120})^2 + 1]} \\ &= 0.09975\end{aligned}$$

$$\begin{aligned}\mu_y &= ln\mu_x - \frac{1}{2}\sigma_y^2 \\ &= ln120 - \frac{1}{2} \times 0.09975^2 \\ &= 4.7825\end{aligned}$$

(2) Compute the density function value at x^*

$$\begin{aligned}f_x(x^*) &= \frac{1}{\sqrt{2\pi}x^*\sigma_y} exp[-\frac{1}{2}(\frac{lnx^* - \mu_y}{\sigma_y})^2] \\ &= \frac{1}{\sqrt{2\pi} \times 80.0402 \times 0.09975} exp[-\frac{1}{2}(\frac{ln80.0402 - 4.7825}{0.09975})^2] \\ &= 1.6114 \times 10^{-5}\end{aligned}$$

(3) Compute the cumulative distribution function value at x^*

From the density function above, it is obvious that the cumulative distribution function can be given as

$$F_x(x^*) = \Phi(\frac{lnx^* - \mu_y}{\sigma_y})$$

(4) Compute $\Phi^{-1}[F_x(x^*)]$

$$\begin{aligned}\Phi^{-1}[F_x(x^*)] &= \frac{lnx^* - \mu_y}{\sigma_y} \\ &= \frac{ln80.0402 - 4.7825}{0.09975} \\ &= -4.0098\end{aligned}$$

(5) Compute $\phi(\Phi^{-1}[F_x(x^*)])$

$$\phi(\Phi^{-1}[F_x(x^*)]) = \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx^* - \mu_y}{\sigma_y})^2]$$

$$= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln80.0402 - 4.7825}{0.09975})^2]$$
$$= 1.2865 \times 10^{-4}$$

(6) Compute the mean value and standard deviation of the equivalent normal distribution, $\mu_{x'}$ and $\sigma_{x'}$ using Eqs. (3.3.39a) and (3.3.39b)

$$\sigma_{x'} = \frac{\phi(\Phi^{-1}[F_x(x^*)])}{f_x(x^*)}$$
$$= \frac{1.2865 \times 10^{-4}}{1.6114 \times 10^{-5}}$$
$$= 7.9841$$

$$\mu_{x'} = x^* - \Phi^{-1}[F_x(x^*)]\sigma_{x'}$$
$$= 80.0402 + 4.0098 \times 7.9841$$
$$= 112.0553$$

Using $\mu_{x'}$ and $\sigma_{x'}$, the standard normal variable of x can be computed from Eq. (3.3.38b).

Example 3.5

x is a random variable having a type-I extreme value distribution. Its mean value and standard deviation are $\mu_x = 4$ and $\sigma_x = 1$, respectively. Calculate the mean value and standard deviation of the equivalent normal distribution variable x' at $x^* = \mu_x = 4$.

(1) Compute the scale and location parameters, δ and t_0, of the type-I extreme value distribution for the variable x using Eq. (1.65). In the following expression, μ_P, σ_P, δ, and t_0 are equivalent to μ_x, α_x, β, and $1/\alpha$ given in Eq. (1.65), respectively.

$$t_0 = \frac{\sigma_x}{1.2825}$$

$$= \frac{1}{1.2825}$$
$$= 0.7797$$

$$\delta = \mu_x - 0.5772 t_0$$
$$= 4 - 0.5772 \times 0.7797$$
$$= 3.5499$$

(2) Compute the density function value at x^*

$$f_x(x^*) = \frac{1}{t_0} exp\{-(\frac{x^* - \delta}{t_0}) - exp[-(\frac{x^* - \delta}{t_0})]\}$$
$$= 1.2825 \times exp\{-(\frac{4 - 3.5499}{0.7797}) - exp[-(\frac{4 - 3.5499}{0.7797})]\}$$
$$= 0.4107$$

(3) Compute the cumulative distribution function value at x^*

$$F_x(x^*) = exp\{-exp(-\frac{x^* - \delta}{t_0})\}$$
$$= exp\{-exp(-\frac{4 - 3.5499}{0.7797})\}$$
$$= 0.5704$$

(4) Compute $\Phi^{-1}[F_x(x^*)]$

$$\Phi^{-1}[F_x(x^*)] = \Phi^{-1}[0.5704]$$
$$= 0.177$$

(5) Compute $\phi(\Phi^{-1}[F_x(x^*)])$

$$\begin{aligned}\phi(\Phi^{-1}[F_x(x^*)]) &= \phi(0.177)\\ &= 0.3927\end{aligned}$$

(6) Compute the mean value and standard deviation of the equivalent normal distribution, $\mu_{x'}$ and $\sigma_{x'}$ using Eqs. (3.3.39a) and (3.3.39b)

$$\begin{aligned}\sigma_{x_i'} &= \frac{\phi(\Phi^{-1}[F_{x_i}(x_i^*)])}{f_{x_i}(x_i^*)}\\ &= \frac{0.3927}{0.4107}\\ &= 0.9561\end{aligned}$$

$$\begin{aligned}\mu_{x_i'} &= x_i^* - \Phi^{-1}[F_{x_i}(x_i^*)]\sigma_{x_i'}\\ &= 4 - \Phi^{-1}[0.5704] \times 0.9561\\ &= 3.8305\end{aligned}$$

Example 3.6

x is a random variable having a Weibull distribution. Its mean value and standard deviation are $\mu_x = 50$ and $\sigma_x = 5$, respectively. Calculate the mean value and standard deviation of the equivalent normal distribution variable x' at $x^* = \mu_x = 50$.

(1) Compute the parameters, α and β, of the Weibull distribution for the variable x using Eqs. (1.58) and (1.59).

$$\frac{5}{50} = [\frac{\Gamma(\frac{2}{\alpha}+1)}{\Gamma^2(\frac{1}{\alpha}+1)} - 1]^{0.5}$$

$$\alpha = 12.1534$$

$$\begin{aligned}\beta &= \frac{\mu_x}{\Gamma(\frac{1}{\alpha}+1)} \\ &= \frac{50}{\Gamma(\frac{1}{12.1534}+1)} \\ &= 52.1519\end{aligned}$$

Note: If Eq. (1.60) is used, the approximate parameters can be solved as below:

$$\alpha = C_x^{-1.08} = (\frac{5}{50})^{-1.08} = 12.0226$$

$$\beta = \frac{\mu_x}{\Gamma(\frac{1}{\alpha}+1)} = \frac{50}{\Gamma(\frac{1}{12.0226}+1)} = 52.1728$$

The approximate parameters are very close to the exact values, so this method can be used for practical engineering problems.

(2) Compute the density function value at x^*

$$\begin{aligned}f_x(x^*) &= \frac{\alpha x^{\alpha-1}}{\beta^\alpha} exp[-(\frac{x}{\beta})^\alpha] \\ &= \frac{12.1534 \times 50^{12.1534-1}}{52.1519^{12.1534}} exp[-(\frac{50}{52.1519})^{12.1534}] \\ &= 7.9998^{-2}\end{aligned}$$

(3) Compute the cumulative distribution function value at x^*

$$\begin{aligned}F_x(x^*) &= 1 - exp[-(\frac{x}{\beta})^\alpha] \\ &= 1 - exp[-(\frac{50}{52.1519})^{12.1534}] \\ &= 0.4508\end{aligned}$$

(4) Compute $\Phi^{-1}[F_x(x^*)]$

$$\Phi^{-1}[F_x(x^*)] = \Phi^{-1}[0.4508]$$
$$= -0.1237$$

(5) Compute $\phi(\Phi^{-1}[F_x(x^*)])$

$$\phi(\Phi^{-1}[F_x(x^*)]) = \phi(-0.1237)$$
$$= 0.3959$$

(6) Compute the mean value and standard deviation of the equivalent normal distribution, $\mu_{x'}$ and $\sigma_{x'}$ using Eqs. (3.3.39a) and (3.3.39b)

$$\sigma_{x_i'} = \frac{\phi(\Phi^{-1}[F_{x_i}(x_i^*)])}{f_{x_i}(x_i^*)}$$
$$= \frac{0.3959}{7.9998^{-2}}$$
$$= 4.9489$$

$$\mu_{x_i'} = x_i^* - \Phi^{-1}[F_{x_i}(x_i^*)]\sigma_{x_i'}$$
$$= 50 - \Phi^{-1}[0.4508] \times 4.9489$$
$$= 50.6123$$

3.3.2.6 RF/HL-RF algorithm

The RF algorithm is similar to the Hasofer-Lind iteration method shown in section 3.3.2.2, except that the steps 2 and 4 are necessary to implement the calculation of the mean and standard deviation of the equivalent normal variables based on Eqs. (3.3.39a) and (3.3.39b).

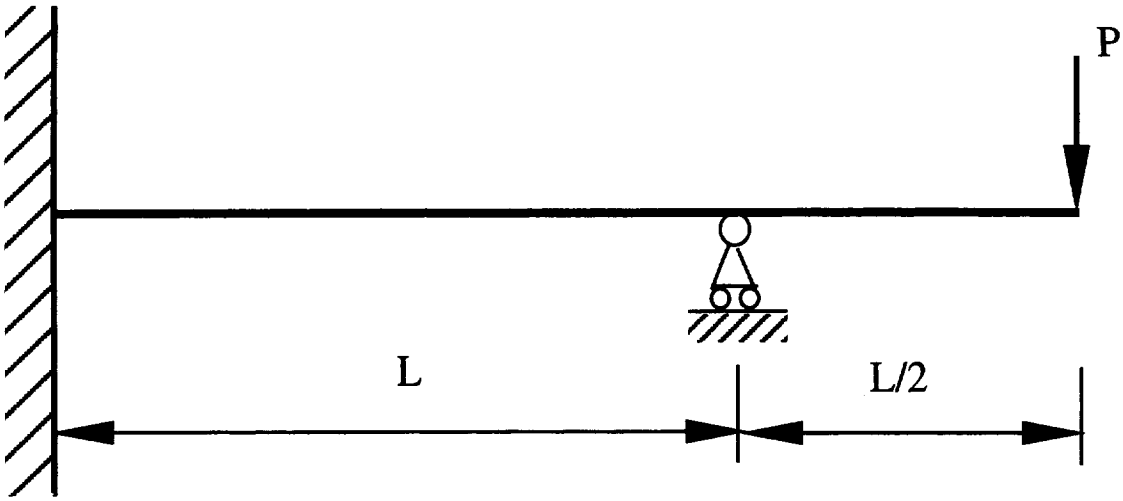


Fig. 3.10 Beam with a Concentrated Load

The RF method can be also called HL-RF method since the iteration algorithm was originally proposed by Hasofer and Lind and later extended by Rackwitz and Fiessler to include random variable distribution information. The following example is given to explain the HL-RF algorithm.

Example 3.7

Consider the plane frame structure shown in Fig. 3.10. Evaluate the safety index β and the coordinates x_i^* of MPP. The failure principle of the displacement is

$$d_{max} = \frac{5PL^3}{48EI} \geq \frac{L}{30}$$

where d_{max} is an allowable maximum displacement, E is Young's modulus, and I is the bending inertial moment of the cross section. The mean values, μ_P, μ_L, μ_E, μ_I, of the load P, the beam

length L, the Young's modulus E, and the inertial moment I are 4 kN, 5 m, $2.0 \times 10^7 kN/m^2$, and $10^{-4}m^4$, respectively. The corresponding standard deviations, σ_P, σ_L, σ_E, σ_I, are 1 kN, 0 m, $0.5 \times 10^7 kN/m^2$, and $0.2 \times 10^{-4}m^4$. Because $\sigma_L = 0$, L is a deterministic parameter (L=5 m). There are three random variables in this example. P is the type-I extreme value distribution, and E and I are normal distribution. The limit state function is given as

$$EI - 78.12P = 0$$

(1) Compute the scale and location parameters of the type-I extreme value distribution using Eq. (1.65) for the variable P. In the following expression, μ_P, σ_P, and δ are equivalent to μ_x, α_x and β given in Eq. (1.65), respectively.

$$\mu_P = \delta + 0.5772/\alpha$$

$$\sigma_P = 1.2825/\alpha$$

Substituting $\mu_P = 4kN$ and $\sigma_P = 1kN$ into the above formulas, the scale and location parameters are obtained as

$$\delta = 3.5499, \qquad \alpha = 1.2825$$

(2) Iteration 1:

(a) Compute the mean and standard deviation of the equivalent normal distribution for P:

First, assuming the design point, $X^* = \{E^*, I^*, P^*\}^T$, as the mean value point, the coordinates of the initial design point are

$$E^* = \mu_E = 2 \times 10^7, \qquad I^* = \mu_I = 10^{-4}, \qquad P^* = \mu_P = 4$$

The density function value at P^* is

$$f_P(P^*) = \alpha exp\{-(P^* - \delta)\alpha - exp[-(P^* - \delta)\alpha]\}$$

$$= 1.2825 \cdot exp\{-(4 - 3.5499) \times 1.2825 - exp[-(4 - 3.5499) \times 1.2825]\}$$

$$= 0.4107$$

The cumulative distribution function value at P^* is

$$\begin{aligned} F_P(P^*) &= exp\{-exp[-(P^* - \delta)\alpha]\} \\ &= exp\{-exp[-(4 - 3.5499) \times 1.2825]\} \\ &= 0.5704 \end{aligned}$$

Therefore the standard deviation and mean value of the equivalent normal variable at P^* from Eqs. (3.3.44a) and (3.3.44b) are

$$\begin{aligned} \sigma_{P'} &= \frac{\phi(\Phi^{-1}[F_P(P^*)])}{f_P(P^*)} \\ &= \frac{\phi(\Phi^{-1}[0.5703])}{0.4107} \\ &= \frac{0.3927}{0.4107} \\ &= 0.9561 \end{aligned}$$

where $\Phi^{-1}[0.5704] = 0.177$ and $\phi(0.177) = 0.3927$.

$$\begin{aligned} \mu_{P'} &= P^* - \Phi^{-1}[F_P(P^*)]\sigma_{P'} \\ &= 4 - \Phi^{-1}[0.5704] \times 0.9561 \\ &= 3.8304 \end{aligned}$$

(b) Compute the function value and gradients of the limit state function at the mean value point:

$$
\begin{aligned}
g(E^*, I^*, P^*) &= EI - 78.12P \\
&= 2 \times 10^7 \times 10^{-4} - 78.12 \times 4 \\
&= 1687.52
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial g(X^*)}{\partial E} &= 10^{-4} \\
\frac{\partial g(X^*)}{\partial I} &= 2 \times 10^7 \\
\frac{\partial g(X^*)}{\partial P} &= -78.12
\end{aligned}
$$

(c) Compute the initial β using the mean value method and its direction cosine α_i

$$
\begin{aligned}
\beta_1 &= \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} \\
&= \frac{g(E^*, I^*, P^*)}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= \frac{1687.52}{\sqrt{(10^{-4} \times 0.5 \times 10^7)^2 + (2 \times 10^7 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 0.9561)^2}} \\
&= 2.6383
\end{aligned}
$$

$$
\begin{aligned}
\alpha_E &= -\frac{\frac{\partial g}{\partial E}|_\mu \sigma_E}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{10^{-4} \times 0.5 \times 10^7}{\sqrt{(10^{-4} \times 0.5 \times 10^7)^2 + (2 \times 10^7 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 0.9561)^2}} \\
&= -0.7756
\end{aligned}
$$

$$
\alpha_I = -\frac{\frac{\partial g}{\partial I}|_\mu \sigma_I}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}}
$$

$$= -\frac{2 \times 10^7 \times 0.2 \times 10^{-4}}{\sqrt{(10^{-4} \times 0.5 \times 10^7)^2 + (2 \times 10^7 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 0.9561)^2}}$$
$$= -0.6205$$

$$\alpha_P = -\frac{\frac{\partial g}{\partial P}|_\mu \sigma_P}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}}$$
$$= -\frac{-78.12 \times 0.9561}{\sqrt{(10^{-4} \times 0.5 \times 10^7)^2 + (2 \times 10^7 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 0.9561)^2}}$$
$$= 0.1159$$

(d) Compute the coordinates of the new design point from Eq.(3.3.28):

$$E^* = \mu_E + \beta_1 \sigma_E \alpha_E$$
$$= 2 \times 10^7 + 2.6383 \times 0.5 \times 10^7 \times (-0.7756)$$
$$= 9768710.0966$$

$$I^* = \mu_I + \beta_1 \sigma_I \alpha_I$$
$$= 10^{-4} + 2.6383 \times 0.2 \times 10^{-4} \times (-0.6205)$$
$$= 0.6726 \times 10^{-4}$$

$$P^* = \mu_{P'} + \beta_1 \sigma_{P'} \alpha_P$$
$$= 3.8304 + 2.6383 \times 0.9561 \times 0.1159$$
$$= 4.1227$$

$$u_E^* = \frac{E^* - \mu_E}{\sigma_E} = \frac{9768710.0966 - 2 \times 10^7}{0.5 \times 10^7} = -2.0463$$

$$u_I^* = \frac{I^* - \mu_I}{\sigma_I} = \frac{0.6726 \times 10^{-4} - 10^{-4}}{0.2 \times 10^{-4}} = -1.6370$$

$$u_P^* = \frac{P^* - \mu_{P'}}{\sigma_{P'}} = \frac{4.1227 - 3.8304}{0.9561} = 0.3057$$

(3) Iteration 2:

(a) Compute the mean and standard deviation of the equivalent normal distribution at P^*

The density function value at P^* is

$$\begin{aligned} f_P(P^*) &= \alpha exp\{-(P^* - \delta)\alpha - exp[-(P^* - \delta)\alpha]\} \\ &= 1.2825 \cdot exp\{-(4.1227 - 3.5499) \times 1.2825 - exp[-(4.1227 - 3.5499) \times 1.2825]\} \\ &= 0.3808 \end{aligned}$$

The cumulative distribution function value at P^* is

$$\begin{aligned} F_P(P^*) &= exp\{-exp[-(P^* - \delta)\alpha]\} \\ &= exp\{-exp[-(4.1227 - 3.5499) \times 1.2825]\} \\ &= 0.6189 \end{aligned}$$

The standard deviation and mean value of the equivalent normal variable at P^* are

$$\begin{aligned} \sigma_{P'} &= \frac{\phi(\Phi^{-1}[F_P(P^*)])}{f_P(P^*)} \\ &= \frac{\phi(\Phi^{-1}[0.6189])}{0.3808} \\ &= \frac{0.3811}{0.3808} \\ &= 1.0007 \end{aligned}$$

where $\Phi^{-1}[0.6189] = 0.3028$ and $\phi(0.3028) = 0.3811$.

$$\begin{aligned}\mu_{P'} &= P^* - \Phi^{-1}[F_P(P^*)]\sigma_{P'} \\ &= 4.1227 - \Phi^{-1}[0.6189] \times 1.0007 \\ &= 3.8197\end{aligned}$$

(b) Compute the function value and gradients of the limit state function at $X^*(E^*, I^*, P^*)$

$$\begin{aligned}g(E^*, I^*, P^*) &= EI - 78.12P \\ &= 9768710.0966 \times 0.6726 \times 10^{-4} - 78.12 \times 4.1227 \\ &= 334.9737\end{aligned}$$

$$\begin{aligned}\frac{\partial g(X^*)}{\partial E} &= 6.726 \times 10^{-5} \\ \frac{\partial g(X^*)}{\partial I} &= 9768710.0966 \\ \frac{\partial g(X^*)}{\partial P} &= -78.12\end{aligned}$$

(c) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}\beta_2 &= \frac{g(X^*) - \frac{\partial g(X^*)}{\partial E}\sigma_E u_E^* - \frac{\partial g(X^*)}{\partial I}\sigma_I u_I^* - \frac{\partial g(X^*)}{\partial P}\sigma_P u_P^*}{\sqrt{(\frac{\partial g(X^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(X^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(X^*)}{\partial P}\sigma_P)^2}} \\ &= \frac{334.9737 + 672.6 \times 0.5 \times 2.0463 + 976.87 \times 0.2 \times 1.6370 + 78.12 \times 1.0007 \times 0.3057}{\sqrt{(6.726 \times 10^{-5} \times 0.5 \times 10^7)^2 + (9768710.0966 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.0007)^2}} \\ &= 3.4449\end{aligned}$$

$$\begin{aligned}\alpha_E &= -\frac{\frac{\partial g}{\partial E}|_\mu \sigma_E}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\ &= -\frac{6.726 \times 10^{-5} \times 0.5 \times 10^7}{\sqrt{(6.726 \times 10^{-5} \times 0.5 \times 10^7)^2 + (9768710.0966 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.0007)^2}} \\ &= -0.8477\end{aligned}$$

$$\alpha_I = -\frac{\frac{\partial g}{\partial I}|_\mu \sigma_I}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}}$$

$$= -\frac{9768710.0966 \times 0.2 \times 10^{-4}}{\sqrt{(6.726 \times 10^{-5} \times 0.5 \times 10^7)^2 + (9768710.0966 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.0007)^2}}$$

$$= -0.4925$$

$$\alpha_P = -\frac{\frac{\partial g}{\partial P}|_\mu \sigma_P}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}}$$

$$= -\frac{-78.12 \times 1.0007}{\sqrt{(6.726 \times 10^{-5} \times 0.5 \times 10^7)^2 + (9768710.0966 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.0007)^2}}$$

$$= 0.1971$$

(d) Compute the coordinates of the new design point from Eq.(3.3.28):

$$E^* = \mu_E + \beta_2 \sigma_E \alpha_E$$

$$= 2 \times 10^7 + 3.4449 \times 0.5 \times 10^7 \times (-0.8477)$$

$$= 5398459.5316$$

$$I^* = \mu_I + \beta_2 \sigma_I \alpha_I$$

$$= 10^{-4} + 3.4449 \times 0.2 \times 10^{-4} \times (-0.4925)$$

$$= 0.6607 \times 10^{-4}$$

$$P^* = \mu_{P'} + \beta_2 \sigma_{P'} \alpha_P$$

$$= 3.8197 + 3.4449 \times 1.0007 \times 0.1971$$

$$= 4.4990$$

$$u_E^* = \frac{E^* - \mu_E}{\sigma_E} = \frac{5398459.5316 - 2 \times 10^7}{0.5 \times 10^7} = -2.9203$$

$$u_I^* = \frac{I^* - \mu_I}{\sigma_I} = \frac{1.3393 \times 10^{-4} - 10^{-4}}{0.2 \times 10^{-4}} = -1.6966$$

$$u_P^* = \frac{P^* - \mu_{P'}}{\sigma_{P'}} = \frac{4.4990 - 3.8197}{1.0007} = 0.6788$$

(e) Check β convergence

$$\varepsilon = \frac{|\beta_2 - \beta_1|}{\beta_1} = \frac{3.4449 - 2.6383}{2.6383} = 0.3057$$

Since $\varepsilon > \varepsilon_r(0.001)$, continue the process.

(4) Iteration 3:

(a) Compute the mean and standard deviation of the equivalent normal distribution at P^*

The density function value at P^* is

$$\begin{aligned} f_P(P^*) &= \alpha exp\{-(P^* - \delta)\alpha - exp[-(P^* - \delta)\alpha]\} \\ &= 1.2825 \cdot exp\{-(4.4990 - 3.5499) \times 1.2825 - exp[-(4.4990 - 3.5499) \times 1.2825]\} \\ &= 0.2824 \end{aligned}$$

The cumulative distribution function value at P^* is

$$\begin{aligned} F_P(P^*) &= exp\{-exp[-(P^* - \delta)\alpha]\} \\ &= exp\{-exp[-(4.4990 - 3.5499) \times 1.2825]\} \\ &= 0.7438 \end{aligned}$$

The standard deviation and mean value of the equivalent normal variable at P^* are

$$\begin{aligned}
\sigma_{P'} &= \frac{\phi(\Phi^{-1}[F_P(P^*)])}{f_P(P^*)} \\
&= \frac{\phi(\Phi^{-1}[0.7438])}{0.2824} \\
&= \frac{0.3219}{0.2824} \\
&= 1.13998
\end{aligned}$$

where $\Phi^{-1}[0.7438] = 0.65495$ and $\phi(0.65495) = 0.3219$.

$$\begin{aligned}
\mu_{P'} &= P^* - \Phi^{-1}[F_P(P^*)]\sigma_{P'} \\
&= 4.4990 - \Phi^{-1}[0.7438] \times 1.13998 \\
&= 3.7524
\end{aligned}$$

(b) Compute the function value and gradients of the limit state function at $X^*(E^*, I^*, P^*)$

$$\begin{aligned}
g(E^*, I^*, P^*) &= EI - 78.12P \\
&= 5398459.5316 \times 1.3393 \times 10^{-4} - 78.12 \times 4.4990 \\
&= 5.2055
\end{aligned}$$

$$\begin{aligned}
\frac{\partial g(X^*)}{\partial E} &= 6.6069 \times 10^{-5} \\
\frac{\partial g(X^*)}{\partial I} &= 5398459.5316 \\
\frac{\partial g(X^*)}{\partial P} &= -78.12
\end{aligned}$$

(c) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_3 &= \frac{g(X^*) - \frac{\partial g(X^*)}{\partial E}\sigma_E u_E^* - \frac{\partial g(X^*)}{\partial I}\sigma_I u_I^* - \frac{\partial g(X^*)}{\partial P}\sigma_P u_P^*}{\sqrt{(\frac{\partial g(X^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(X^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(X^*)}{\partial P}\sigma_P)^2}} \\
&= \frac{5.2055 + 660.69 \times 0.5 \times 2.9203 + 539.85 \times 0.2 \times 1.6966 + 78.12 \times 1.13998 \times 0.6788}{\sqrt{(6.6069 \times 10^{-5} \times 0.5 \times 10^7)^2 + (5398459.5316 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.13998)^2}} \\
&= 3.3766
\end{aligned}$$

$$\begin{aligned}
\alpha_E &= -\frac{\frac{\partial g}{\partial E}|_\mu \sigma_E}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{6.6069 \times 10^{-5} \times 0.5 \times 10^7}{\sqrt{(6.6069 \times 10^{-5} \times 0.5 \times 10^7)^2 + (5398459.5316 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.13998)^2}} \\
&= -0.9208
\end{aligned}$$

$$\begin{aligned}
\alpha_I &= -\frac{\frac{\partial g}{\partial I}|_\mu \sigma_I}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{5398459.5316 \times 0.2 \times 10^{-4}}{\sqrt{(6.6069 \times 10^{-5} \times 0.5 \times 10^7)^2 + (5398459.5316 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.13998)^2}} \\
&= -0.3009
\end{aligned}$$

$$\begin{aligned}
\alpha_P &= -\frac{\frac{\partial g}{\partial P}|_\mu \sigma_P}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{-78.12 \times 1.13998}{\sqrt{(6.6069 \times 10^{-5} \times 0.5 \times 10^7)^2 + (5398459.5316 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.13998)^2}} \\
&= 0.2482
\end{aligned}$$

(d) Compute the coordinates of the new design point from Eq.(3.3.28):

$$\begin{aligned}
E^* &= \mu_E + \beta_3 \sigma_E \alpha_E \\
&= 2 \times 10^7 + 3.3766 \times 0.5 \times 10^7 \times (-0.9208) \\
&= 4454699.4278
\end{aligned}$$

$$
\begin{aligned}
I^* &= \mu_I + \beta_3 \sigma_I \alpha_I \\
&= 10^{-4} + 3.3766 \times 0.2 \times 10^{-4} \times (-0.3009) \\
&= 0.7968 \times 10^{-4}
\end{aligned}
$$

$$
\begin{aligned}
P^* &= \mu_P + \beta_3 \sigma_P \alpha_P \\
&= 3.7524 + 3.3766 \times 1.13998 \times 0.2482 \\
&= 4.7079
\end{aligned}
$$

$$
u_E^* = \frac{E^* - \mu_E}{\sigma_E} = \frac{4454699.4278 - 2 \times 10^7}{0.5 \times 10^7} = -3.1091
$$

$$
u_I^* = \frac{I^* - \mu_I}{\sigma_I} = \frac{0.7968 \times 10^{-4} - 10^{-4}}{0.2 \times 10^{-4}} = -1.0162
$$

$$
u_P^* = \frac{P^* - \mu_{P'}}{\sigma_{P'}} = \frac{4.7079 - 3.7524}{1.13998} = 0.8381
$$

(e) Check β convergence

$$
\varepsilon = \frac{|\beta_3 - \beta_2|}{\beta_2} = \frac{|3.3766 - 3.4449|}{3.4449} = 0.0198
$$

Since $\varepsilon > \varepsilon_r(0.001)$, continue the process.

(5) Iteration 4:

(a) Compute the mean and standard deviation of the equivalent normal distribution at P^*

The density function value at P^* is

$$
\begin{aligned}
f_P(P^*) &= \alpha exp\{-(P^* - \delta)\alpha - exp[-(P^* - \delta)\alpha]\} \\
&= 1.2825 \cdot exp\{-(4.7079 - 3.5499) \times 1.2825 - exp[-(4.7079 - 3.5499) \times 1.2825]\} \\
&= 0.2316
\end{aligned}
$$

The cumulative distribution function value at P^* is

$$\begin{aligned} F_P(P^*) &= exp\{-exp[-(P^* - \delta)\alpha]\} \\ &= exp\{-exp[-(4.7079 - 3.5499) \times 1.2825]\} \\ &= 0.7973 \end{aligned}$$

The standard deviation and mean value of the equivalent normal variable at P^* are

$$\begin{aligned} \sigma_{P'} &= \frac{\phi(\Phi^{-1}[F_P(P^*)])}{f_P(P^*)} \\ &= \frac{\phi(\Phi^{-1}[0.7973])}{0.2368} \\ &= \frac{0.2822}{0.2368} \\ &= 1.2184 \end{aligned}$$

where $\Phi^{-1}[0.7973] = 0.8321$ and $\phi(0.8321) = 0.2822$.

$$\begin{aligned} \mu_{P'} &= P^* - \Phi^{-1}[F_P(P^*)]\sigma_{P'} \\ &= 4.7079 - \Phi^{-1}[0.7973] \times 1.2184 \\ &= 3.6939 \end{aligned}$$

(b) Compute the function value and gradients of the limit state function at $X^*(E^*, I^*, P^*)$

$$\begin{aligned} g(E^*, I^*, P^*) &= EI - 78.12P \\ &= 4454699.4278 \times 0.7968 \times 10^{-4} - 78.12 \times 4.7079 \\ &= -12.8426 \end{aligned}$$

$$\frac{\partial g(X^*)}{\partial E} = 7.9676 \times 10^{-5}$$
$$\frac{\partial g(X^*)}{\partial I} = 4454699.4278$$
$$\frac{\partial g(X^*)}{\partial P} = -78.12$$

(c) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}
\beta_4 &= \frac{g(X^*) - \frac{\partial g(X^*)}{\partial E}\sigma_E u_E^* - \frac{\partial g(X^*)}{\partial I}\sigma_I u_I^* - \frac{\partial g(X^*)}{\partial P}\sigma_P u_P^*}{\sqrt{(\frac{\partial g(X^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(X^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(X^*)}{\partial P}\sigma_P)^2}} \\
&= \frac{-12.8426 + 796.76 \times 0.5 \times 3.1091 + 445.47 \times 0.2 \times 1.0162 + 78.12 \times 1.2184 \times 0.8381}{\sqrt{(7.9676 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4454699.4278 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.2184)^2}} \\
&= 3.3292
\end{aligned}$$

$$\begin{aligned}
\alpha_E &= -\frac{\frac{\partial g}{\partial E}|_\mu \sigma_E}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{7.9676 \times 10^{-5} \times 0.5 \times 10^7}{\sqrt{(7.9676 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4454699.4278 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.2184)^2}} \\
&= -0.9504
\end{aligned}$$

$$\begin{aligned}
\alpha_I &= -\frac{\frac{\partial g}{\partial I}|_\mu \sigma_I}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{4454699.4278 \times 0.2 \times 10^{-4}}{\sqrt{(7.9676 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4454699.4278 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.2184)^2}} \\
&= -0.2125
\end{aligned}$$

$$\begin{aligned}
\alpha_P &= -\frac{\frac{\partial g}{\partial P}|_\mu \sigma_P}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{-78.12 \times 1.2184}{\sqrt{(7.9676 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4454699.4278 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.2184)^2}} \\
&= 0.2271
\end{aligned}$$

(d) Compute the coordinates of the new design point from Eq.(3.3.28):

$$\begin{aligned} E^* &= \mu_E + \beta_4 \sigma_E \alpha_E \\ &= 2 \times 10^7 + 3.3292 \times 0.5 \times 10^7 \times (-0.9504) \\ &= 4179859.5780 \end{aligned}$$

$$\begin{aligned} I^* &= \mu_I + \beta_4 \sigma_I \alpha_I \\ &= 10^{-4} + 3.3292 \times 0.2 \times 10^{-4} \times (-0.2125) \\ &= 0.8585 \times 10^{-4} \end{aligned}$$

$$\begin{aligned} P^* &= \mu_P + \beta_4 \sigma_P \alpha_P \\ &= 3.6939 + 3.3292 \times 1.2184 \times 0.2271 \\ &= 4.6151 \end{aligned}$$

$$u_E^* = \frac{E^* - \mu_E}{\sigma_E} = \frac{4179859.5780 - 2 \times 10^7}{0.5 \times 10^7} = -3.1640$$

$$u_I^* = \frac{I^* - \mu_I}{\sigma_I} = \frac{0.8585 \times 10^{-4} - 10^{-4}}{0.2 \times 10^{-4}} = -0.7076$$

$$u_P^* = \frac{P^* - \mu_{P'}}{\sigma_{P'}} = \frac{4.6151 - 3.6939}{1.2184} = 0.7560$$

(e) Check β convergence

$$\varepsilon = \frac{|\beta_4 - \beta_3|}{\beta_3} = \frac{3.6939 - 3.3766}{3.3766} = 0.094$$

Since $\varepsilon > \varepsilon_r(0.001)$, continue the process.

(6) Iteration 5:

(a) Compute the mean and standard deviation of the equivalent normal distribution at P^*

The density function value at P^* is

$$\begin{aligned} f_P(P^*) &= \alpha exp\{-(P^* - \delta)\alpha - exp[-(P^* - \delta)\alpha]\} \\ &= 1.2825 \cdot exp\{-(4.6151 - 3.5499) \times 1.2825 - exp[-(4.6151 - 3.5499) \times 1.2825]\} \\ &= 0.2535 \end{aligned}$$

The cumulative distribution function value at P^* is

$$\begin{aligned} F_P(P^*) &= exp\{-exp[-(P^* - \delta)\alpha]\} \\ &= exp\{-exp[-(4.6151 - 3.5499) \times 1.2825]\} \\ &= 0.7748 \end{aligned}$$

The standard deviation and mean value of the equivalent normal variable at P^* are

$$\begin{aligned} \sigma_{P'} &= \frac{\phi(\Phi^{-1}[F_P(P^*)])}{f_P(P^*)} \\ &= \frac{\phi(\Phi^{-1}[0.7748])}{0.2535} \\ &= \frac{0.3000}{0.2535} \\ &= 1.1835 \end{aligned}$$

where $\Phi^{-1}[0.7748] = 0.7548$ and $\phi(0.7548) = 0.3000$.

$$\begin{aligned} \mu_{P'} &= P^* - \Phi^{-1}[F_P(P^*)]\sigma_{P'} \\ &= 4.6151 - \Phi^{-1}[0.7748] \times 1.1835 \\ &= 3.7217 \end{aligned}$$

(b) Compute the function value and gradients of the limit state function at $X^*(E^*, I^*, P^*)$

$$\begin{aligned} g(E^*, I^*, P^*) &= EI - 78.12P \\ &= 4179859.5780 \times 0.8585 \times 10^{-4} - 78.12 \times 4.6151 \\ &= -1.6961 \end{aligned}$$

$$\frac{\partial g(X^*)}{\partial E} = 8.5848 \times 10^{-5}$$
$$\frac{\partial g(X^*)}{\partial I} = 4179859.5779$$
$$\frac{\partial g(X^*)}{\partial P} = -78.12$$

(c) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned} \beta_5 &= \frac{g(X^*) - \frac{\partial g(X^*)}{\partial E}\sigma_E u_E^* - \frac{\partial g(X^*)}{\partial I}\sigma_I u_I^* - \frac{\partial g(X^*)}{\partial P}\sigma_P u_P^*}{\sqrt{(\frac{\partial g(X^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(X^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(X^*)}{\partial P}\sigma_P)^2}} \\ &= \frac{-1.6961 + 858.48 \times 0.5 \times 3.1640 + 417.98 \times 0.2 \times 0.7076 + 78.12 \times 1.1835 \times 0.7560}{\sqrt{(8.5848 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4179859.5779 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1835)^2}} \\ &= 3.3232 \end{aligned}$$

$$\begin{aligned} \alpha_E &= -\frac{\frac{\partial g}{\partial E}|_\mu \sigma_E}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\ &= -\frac{8.5848 \times 10^{-5} \times 0.5 \times 10^7}{\sqrt{(8.5848 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4179859.5779 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1835)^2}} \\ &= -0.9603 \end{aligned}$$

$$\alpha_I = -\frac{\frac{\partial g}{\partial I}|_\mu \sigma_I}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}}$$

$$= -\frac{4179859.5779 \times 0.2 \times 10^{-4}}{\sqrt{(8.5848 \times 10^{-5} \times 0.5 \times 10^{7})^2 + (4179859.5779 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1835)^2}}$$
$$= -0.1870$$

$$\alpha_P = -\frac{\frac{\partial g}{\partial P}|_{\mu}\sigma_P}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}}$$
$$= -\frac{-78.12 \times 1.1835}{\sqrt{(8.5848 \times 10^{-5} \times 0.5 \times 10^{7})^2 + (4179859.5779 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1835)^2}}$$
$$= 0.2068$$

(d) Compute the coordinates of the new design point from Eq.(3.3.28):

$$E^* = \mu_E + \beta_5 \sigma_E \alpha_E$$
$$= 2 \times 10^7 + 3.3232 \times 0.5 \times 10^7 \times (-0.9603)$$
$$= 4043172.7300$$

$$I^* = \mu_I + \beta_5 \sigma_I \alpha_I$$
$$= 10^{-4} + 3.3232 \times 0.2 \times 10^{-4} \times (-0.1870)$$
$$= 0.8757 \times 10^{-4}$$

$$P^* = \mu_P + \beta_5 \sigma_P \alpha_P$$
$$= 3.7217 + 3.3232 \times 1.1835 \times 0.2068$$
$$= 4.5352$$

$$u_E^* = \frac{E^* - \mu_E}{\sigma_E} = \frac{4043172.7300 - 2 \times 10^7}{0.5 \times 10^7} = -3.1914$$

$$u_I^* = \frac{I^* - \mu_I}{\sigma_I} = \frac{0.8757 \times 10^{-4} - 10^{-4}}{0.2 \times 10^{-4}} = -0.6215$$

$$u_P^* = \frac{P^* - \mu_{P'}}{\sigma_{P'}} = \frac{4.5352 - 3.7217}{1.1835} = 0.6874$$

(e) Check β convergence

$$\varepsilon = \frac{|\beta_5 - \beta_4|}{\beta_4} = \frac{|3.3232 - 3.6939|}{3.6939} = 0.1$$

Since $\varepsilon > \varepsilon_r(0.001)$, continue the process.

(7) Iteration 6:

(a) Compute the mean and standard deviation of the equivalent normal distribution at P^*

The density function value at P^* is

$$\begin{aligned} f_P(P^*) &= \alpha exp\{-(P^* - \delta)\alpha - exp[-(P^* - \delta)\alpha]\} \\ &= 1.2825 \cdot exp\{-(4.5352 - 3.5499) \times 1.2825 - exp[-(4.5352 - 3.5499) \times 1.2825]\} \\ &= 0.2732 \end{aligned}$$

The cumulative distribution function value at P^* is

$$\begin{aligned} F_P(P^*) &= exp\{-exp[-(P^* - \delta)\alpha]\} \\ &= exp\{-exp[-(4.5352 - 3.5499) \times 1.2825]\} \\ &= 0.75380 \end{aligned}$$

The standard deviation and mean value of the equivalent normal variable at P^* are

$$\begin{aligned} \sigma_{P'} &= \frac{\phi(\Phi^{-1}[F_P(P^*)])}{f_P(P^*)} \\ &= \frac{\phi(\Phi^{-1}[0.7538])}{0.2732} \end{aligned}$$

$$= \frac{0.3152}{0.2732}$$
$$= 1.1535$$

where $\Phi^{-1}[0.7538] = 0.6865$ and $\phi(0.6865) = 0.3152$.

$$\begin{aligned}\mu_{P'} &= P^* - \Phi^{-1}[F_P(P^*)]\sigma_{P'} \\ &= 4.5352 - \Phi^{-1}[0.7538] \times 1.1535 \\ &= 3.7433\end{aligned}$$

(b) Compute the function value and gradients of the limit state function at $X^*(E^*, I^*, P^*)$

$$\begin{aligned}g(E^*, I^*, P^*) &= EI - 78.12P \\ &= 4043172.7300 \times 0.8757 \times 10^{-4} - 78.12 \times 4.5352 \\ &= -0.2352\end{aligned}$$

$$\frac{\partial g(X^*)}{\partial E} = 8.7569 \times 10^{-5}$$
$$\frac{\partial g(X^*)}{\partial I} = 4043172.7299$$
$$\frac{\partial g(X^*)}{\partial P} = -78.12$$

(c) Compute β using Eq. (3.3.25) and the direction cosine α_i

$$\begin{aligned}\beta_6 &= \frac{g(X^*) - \frac{\partial g(X^*)}{\partial E}\sigma_E u_E^* - \frac{\partial g(X^*)}{\partial I}\sigma_I u_I^* - \frac{\partial g(X^*)}{\partial P}\sigma_P u_P^*}{\sqrt{(\frac{\partial g(X^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(X^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(X^*)}{\partial P}\sigma_P)^2}} \\ &= \frac{-0.2352 + 875.69 \times 0.5 \times 3.1914 + 404.32 \times 0.2 \times 0.621 + 78.12 \times 1.1535 \times 0.6874}{\sqrt{(8.7569 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4043172.7299 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1535)^2}} \\ &= 3.3222\end{aligned}$$

$$\begin{aligned}
\alpha_E &= -\frac{\frac{\partial g}{\partial E}|_\mu \sigma_E}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{8.7569 \times 10^{-5} \times 0.5 \times 10^7}{\sqrt{(8.7569 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4043172.7299 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1535)^2}} \\
&= -0.9638
\end{aligned}$$

$$\begin{aligned}
\alpha_I &= -\frac{\frac{\partial g}{\partial I}|_\mu \sigma_I}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{4043172.7300 \times 0.2 \times 10^{-4}}{\sqrt{(8.7569 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4043172.7299 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1535)^2}} \\
&= -0.1780
\end{aligned}$$

$$\begin{aligned}
\alpha_P &= -\frac{\frac{\partial g}{\partial P}|_\mu \sigma_P}{\sqrt{(\frac{\partial g(E^*,I^*,P^*)}{\partial E}\sigma_E)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial I}\sigma_I)^2 + (\frac{\partial g(E^*,I^*,P^*)}{\partial P}\sigma_P)^2}} \\
&= -\frac{-78.12 \times 1.1535}{\sqrt{(8.7569 \times 10^{-5} \times 0.5 \times 10^7)^2 + (4043172.7299 \times 0.2 \times 10^{-4})^2 + (-78.12 \times 1.1535)^2}} \\
&= 0.1984
\end{aligned}$$

(d) Compute the coordinates of the new design point from Eq.(3.3.28):

$$\begin{aligned}
E^* &= \mu_E + \beta_6 \sigma_E \alpha_E \\
&= 2 \times 10^7 + 3.3222 \times 0.5 \times 10^7 \times (-0.9638) \\
&= 3989701.3284
\end{aligned}$$

$$\begin{aligned}
I^* &= \mu_I + \beta_6 \sigma_I \alpha_I \\
&= 10^{-4} + 3.3222 \times 0.2 \times 10^{-4} \times (-0.1780) \\
&= 0.8817 \times 10^{-4}
\end{aligned}$$

$$
\begin{aligned}
P^* &= \mu_P + \beta_6 \sigma_P \alpha_P \\
&= 3.7433 + 3.3222 \times 1.1535 \times 0.1984 \\
&= 4.5035
\end{aligned}
$$

$$
u_E^* = \frac{E^* - \mu_E}{\sigma_E} = \frac{3989701.3284 - 2 \times 10^7}{0.5 \times 10^7} = -3.2021
$$

$$
u_I^* = \frac{I^* - \mu_I}{\sigma_I} = \frac{0.8817 \times 10^{-4} - 10^{-4}}{0.2 \times 10^{-4}} = -0.5914
$$

$$
u_P^* = \frac{P^* - \mu_{P'}}{\sigma_{P'}} = \frac{4.5035 - 3.7433}{1.1535} = 0.6590
$$

(e) Check β convergence

$$
\varepsilon = \frac{|\beta_6 - \beta_5|}{\beta_5} = \frac{|3.3222 - 3.3232|}{3.3232} = 0.0003
$$

Since $\varepsilon < \varepsilon_r(0.001)$, stop the process.

The safety index β is 3.3222. Since the limit state function value at the MPP X^* is close to zero, this safety index can be considered as the shortest distance from the origin to the limit state surface.

A computer program based on the RF algorithm is developed to perform the reliability analysis. The flow-chart of the program is given in Fig. 3.11.

3.3.3 Efficient Safety Index Algorithms Using Approximations

Based on the linear, conservative, TANA [8] and TANA2 [10] approximations described in the previous section, the following safety index algorithms and a corresponding computer code AURORA (Approximations Used to Rapidly Obtain Reliability Analysis) have been developed. The flow-chart of the safety index algorithms using approximations is shown in Fig. 3.12.

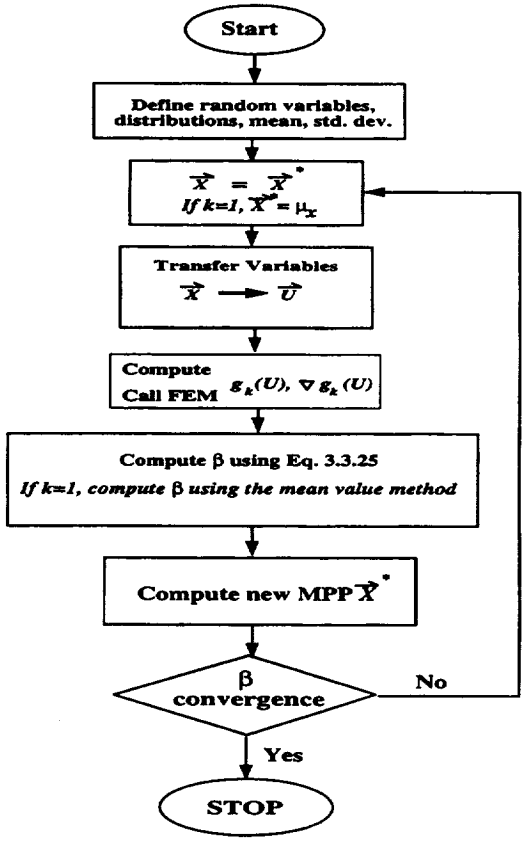


Fig. 3.11 Flow–chart of RF/HL–RF Method

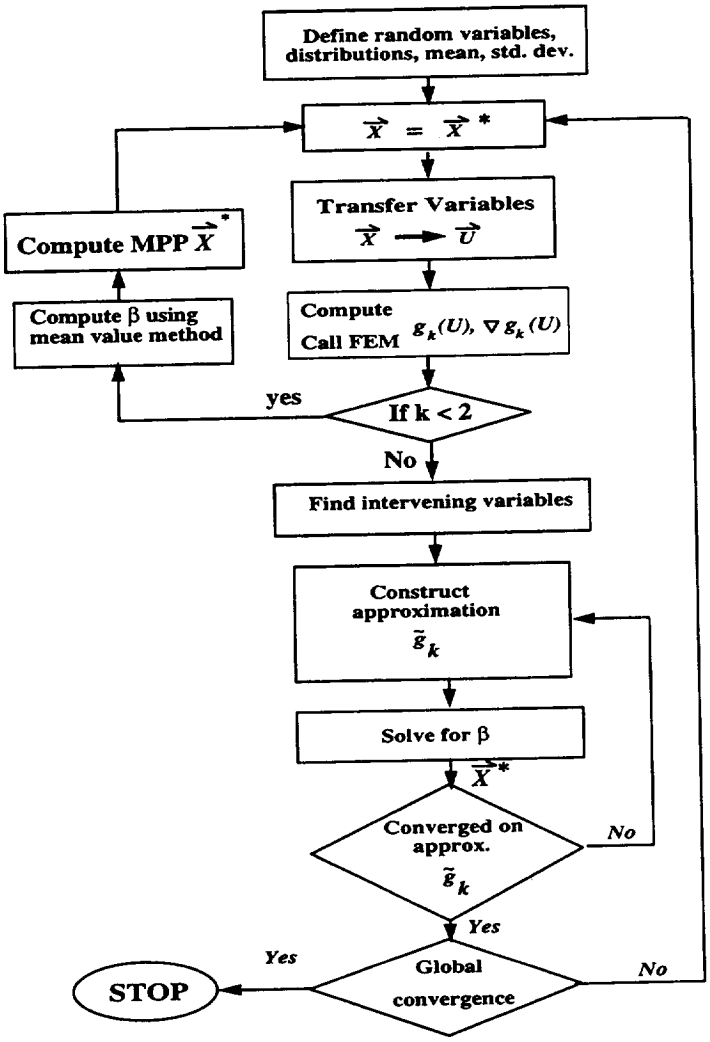


Fig. 3.12 Flow–chart of the Safety Index Algorithms using Approximations

3.3.3.1 Approximate Limit State Function Using Linear Approximation (RF/HL-RF Method)

The algorithm based on the linear approximation is the same as the RF method. The limit state function is approximated as Eq. (3.3.23) at the MPP, and the safety index is obtained by solving iteratively the reliability problem given in Eq. (3.3.20). More details are given in Sections 3.3.2.2 and 3.3.2.6.

3.3.3.2 Approximate Limit State Function Using Conservative Approximation

In the previous algorithm, the hypersurface $G(U)$ was approached by the first-order Taylor expansion at the MPP. For nonlinear problems, this approach is only an approximation, and several iterations are usually required. How fast the algorithm converges depends on how well the linearized limit state function approximates the nonlinear function $g(U)$. Since the conservative approximation has the advantage of including the linear and reciprocal approximations in the formulation, a safety index algorithm using conservative approximation is presented in AURORA.

The limit state function is approximated by the conservative approximation of Eqs. (2.3.7) and (2.3.8). The safety index model of Eq. (3.3.20) becomes:

$$Minimize: \qquad \beta(U) = (U^T U)^{1/2} \tag{3.3.45a}$$

$$Subject \quad to: \qquad \tilde{g}(U) = g(U_k) + \sum_{i=1}^{n} C_i \frac{\partial g(U_k)}{\partial x_i}(x_i - x_{i,k}) = 0 \tag{3.3.45b}$$

The main steps of this algorithm are summarized below:

1). For the first iteration, construct a linear approximation of Eq. (3.3.23) by using the first-order Taylor's series expansion about the mean values, μ;

2). Compute the most probable failure point X_k and safety index β_k using the HL-RF method in U-space, U;

3). Obtain the nonlinear approximation of Eq. (3.3.45b)

4). Find the most probable failure point X_{k+1} of the nonlinear approximation function $\tilde{g}(U)$

and safety index β_{k+1} using the HL-RF method or DOT (Design Optimization Tools) [11], and denote X_{k+1} as the current point and X_k as the previous point;

5). Check the convergence

$$\varepsilon = |\frac{\beta_{k+1} - \beta_k}{\beta_k}|$$

6). Go to step (3) and repeat the process until ε is less than the allowable value.

In step (4), the safety index β can be easily obtained only by computing the explicit function $\tilde{g}(U)$ given in Equation (3.3.45b), in which any optimization scheme or iteration algorithm can be used. The computation of the exact performance function $\tilde{g}(U)$ is not required; therefore , the computer time can be reduced for problems involving complex and implicit performance functions if the conservative approximation provides good accuracy.

3.3.3.3 Approximate Limit State Function Using Two-point Adaptive Nonlinear Approximation (TANA)

The previous algorithms 1 and 2 both used the fixed approximation models to represent the limit state functions. Even though the conservative approximation can be either a linear or reciprocal approximation, the truncation errors of the approximations might be so large that the algorithm may converge very slowly or even result in divergence for some complicated highly nonlinear problems. Therefore, the use of adaptive approximations for different types of problems is necessary and important. The adaptive approximations are constructed by using the first-order Taylor series expansion in terms of adaptive intervening variables. The nonlinearity of the adaptive approximations is automatically changed by using the known information generated during the iteration process.

Based on the two-point adaptive nonlinear approximation (TANA) given in Eq. (2.4.3), the safety index model of Eq. (3.3.20) is obtained as

$$Minimize: \qquad \beta(U) = (U^T U)^{1/2} \tag{3.3.46a}$$

$$Subject \quad to: \qquad \tilde{g}(U) = 0 \tag{3.3.46b}$$

where $\tilde{g}(U)$ is the approximate U-space limit state surface and computed as follows. First, the adaptive approximate limit state surface in X-space is obtained as

$$\tilde{g}(X) = g(X_k) + \frac{1}{r}\sum_{i=1}^{n} x_{i,k}^{1-r}\frac{\partial g(X_k)}{\partial x_i}(x_i^r - x_{i,k}^r) \tag{3.3.47}$$

then, $\tilde{g}(X)$ is mapped into $\tilde{g}(U)$ by using the standard normal or equivalent normal transformations,

$$\tilde{g}(U) = \tilde{g}(\sigma_{x_1'}u_1 + \mu_{x_1'}, \sigma_{x_2'}u_2 + \mu_{x_2'}, ..., \sigma_{x_n'}u_n + \mu_{x_n'}) \tag{3.3.48}$$

The nonlinear index r in Eq.(3.3.47) can be determined from Eq. (2.4.2), *i.e*,

$$g(X_{k-1}) - \{g(X_k) + \frac{1}{r}\sum_{i=1}^{n} x_{i,k}^{1-r}\frac{\partial g(X_k)}{\partial x_i}(x_{i,k-1}^r - x_{i,k}^r)\} = 0 \tag{3.3.49}$$

In AURORA, r is numerically calculated by minimizing the difference between the exact and approximate limit state function at the previous point X_{k-1}. In theory, r can be any positive or negative real number (not equal to 0). r is restricted from -5 to 5 for the X-space iterations in AURORA. The X-space and U-space iterations given in AURORA represent the safety index algorithms in X-space and U-space, respectively. In X-space iterations, the limit state function is approximated in the original space of the random variables (X-space), then the approximation is transferred into U-space. In Y-space iterations, the limit state function is directly approximated in U-space. The details of X-space and U-space iterations are given later in this section. The iteration searching for r starts from $r = 1$. When r is increased or decreased a step length (0.1), the difference ε between the exact and approximate function is calculated. If ε is smaller than the initial error (e.g. corresponding to $r = 1$), the above

iteration is repeated until the allowable error $\varepsilon = 0.001$ or limitation of r is reached, and the nonlinear index r is determined. Otherwise, r is decreased by a half and the above iteration process is repeated until the final r is obtained. This search is computationally inexpensive because Eq. (3.3.49) is available in closed form and very easy to implement.

Usually, the adaptive safety index algorithm is better than the RF method because the nonlinear index r is determined by comparing the linear approximation (starting from 1) and minimizing the difference between the exact and approximate limit state functions. In the process of searching for r, the nonlinear index will automatically become 1 if other values of r can not provide any improvement over the linear approximation.

The main steps of this algorithm are summarized as follows:

1). In the first iteration, compute the mean and standard deviation of the equivalent normal distribution at the mean value point for nonnormal distribution variables. Construct a linear approximation of Eq. (3.3.23) by using the first-order Taylor's series expansion at an initial point (if the initial point is selected as the mean value point, μ, the linear approximation is expanded at μ), and compute the limit state function value and gradients at the initial point;

2). Compute the initial safety index β_1 using the HL-RF method and its direction cosine α_i (if the initial point is the mean value point, the mean value method is used);

3). Compute the new design point using Eq. (3.3.28), X_k;

4). Compute the mean and standard deviation of the equivalent normal distribution at X_k for nonnormal distribution variables. Calculate the limit state function value and gradients at the new design point, X_k;

5). Determine the nonlinearity index r by solving Eq. (3.3.49) based on the information of the current and previous points (when k equals to 2, previous design point is the mean value $\bar{X}$);

6). Obtain the adaptive nonlinear approximation of Eq.(3.3.47);

7). Transform the X-space approximate limit state function into the U-space function using

Eq. (3.3.48);

8). Find the most probable failure point X_{k+1} of the approximate safety index model given in Eq. (3.3.46) using the HL-RF method or DOT and compute the safety index β_{k+1};

9). Check the convergence

$$\varepsilon = |\frac{\beta_{k+1} - \beta_k}{\beta_k}|$$

10). Stop the process if ε satisfies the required convergence tolerance (0.001), otherwise, Continue;

11). Compute the exact limit state function value and approximate gradients at X_{k+1}, and estimate the approximate safety index $\tilde{\beta}_{k+1}$ using the HL-RF method;

12). Approximate β convergence check

$$\varepsilon = \frac{|\tilde{\beta}_{k+1} - \beta_k|}{\beta_k}$$

13). Continue the process if ε satisfies the required convergence tolerance (0.001), otherwise, stop;

14). Compute the exact gradients of the limit state function at X_{k+1} and go to step 5); repeat the process until β converges.

In step (8), the safety index β of the approximate model given in Eq. (3.3.46) can be easily obtained only by computing the explicit function $\tilde{g}(U)$, in which any optimization scheme or iteration algorithm can be used. The computation of the exact performance function $g(X)$ is not required; therefore , the computer time is greatly reduced for problems involving complex and implicit performance functions, particularly with finite element models.

Example 3.8

This example is the same as Example 3.3b. The safety index algorithm using TANA is used to solve β.

(1) Iteration 1:

(a) Set the mean value point as an initial design point and required β convergence tolerance as $\varepsilon_r = 0.001$. Compute the limit state function value and gradients at the mean value point

$$\begin{aligned} g(X^*) &= g(\mu_{x_1}, \mu_{x_2}) = \mu_{x_1}^3 + \mu_{x_2}^3 - 18 \\ &= 10.0^3 + 9.9^3 - 18 \\ &= 1952.299 \end{aligned}$$

$$\frac{\partial g}{\partial x_1}|_\mu = 3\mu_{x_1}^2 = 3 \times 10^2 = 300$$
$$\frac{\partial g}{\partial x_2}|_\mu = 3\mu_{x_2}^2 = 3 \times 9.9^2 = 294.03$$

(b) Compute the initial β using the mean value method and its direction cosine α_i

$$\begin{aligned} \beta_1 &= \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} \\ &= \frac{g(X^*)}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= \frac{1952.299}{\sqrt{(300 \times 5.0)^2 + (294.03 \times 5.0)^2}} \\ &= 0.9295 \end{aligned}$$

$$\begin{aligned} \alpha_1 &= -\frac{\frac{\partial g}{\partial x_1}|_\mu \sigma_{x_1}}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= -\frac{300 \times 5.0}{\sqrt{(300 \times 5.0)^2 + (294.03 \times 5.0)^2}} \\ &= -0.7142 \end{aligned}$$

$$\begin{aligned}\alpha_2 &= -\frac{\frac{\partial g}{\partial x_2}|_\mu \sigma_{x_2}}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2})^2}} \\ &= -\frac{294.03 \times 5.0}{\sqrt{(300 \times 5.0)^2 + (294.03 \times 5.0)^2}} \\ &= -0.6999\end{aligned}$$

(c) Compute a new design point X^* from Eq. (3.3.28)

$$\begin{aligned}x_1^* &= \mu_{x_1} + \beta_1 \sigma_{x_1} \alpha_1 \\ &= 10.0 + 0.9295 \times 5.0 \times (-0.7142) \\ &= 6.6808\end{aligned}$$

$$\begin{aligned}x_2^* &= \mu_{x_2} + \beta_1 \sigma_{x_2} \alpha_2 \\ &= 9.9 + 0.9295 \times 5.0 \times (-0.6999) \\ &= 6.6468\end{aligned}$$

$$u_1^* = \frac{x_1^* - \mu_{x_1}}{\sigma_{x_1}} = \frac{6.6808 - 10.0}{5.0} = -0.6638$$
$$u_2^* = \frac{x_2^* - \mu_{x_2}}{\sigma_{x_2}} = \frac{6.6468 - 9.9}{5.0} = -0.6506$$

(2) Iteration 2:

(a) Compute the limit state function value and gradients at X^*

$$\begin{aligned}g(X^*) &= g(x_1^*, x_2^*) = x_1^{*3} + x_2^{*3} - 18 \\ &= 6.6808^3 + 6.6468^3 - 18 \\ &= 573.8398\end{aligned}$$

$$\frac{\partial g}{\partial x_1}|_\mu = 3x_1^{*2} = 3 \times 6.6808^2 = 133.8982$$
$$\frac{\partial g}{\partial x_2}|_\mu = 3x_2^{*2} = 3 \times 6.6468^2 = 132.5409$$

(b) Compute the nonlinearity index r based on the function values and gradients of the two points $\mu(10.0, 9.9)$ and $X^*(6.6808, 6.6468)$ using Eq. (3.3.49), that is

$$\begin{aligned} g(X_{k-1}) - \{g(X_k) \quad + \quad & \frac{1}{r}\sum_{i=1}^{2} x_{i,k}^{1-r}\frac{\partial g(X_k)}{\partial x_i}(x_{i,k-1}^r - x_{i,k}^r)\} \\ = \quad & 1952.299 - \{573.8398 + \frac{1}{r}\cdot \\ & [6.6808^{1-r} \times 133.8982 \times (10^r - 6.6808^r) + \\ & 6.6468^{1-r} \times 132.5409 \times (9.9^r - 6.6468^r)] \\ \leq \quad & 0.001 \end{aligned}$$

where X_{k-1}=$\mu(10.0, 9.9)$ and X_k=$X^*(6.6808, 6.6468)$.

Using the adaptive search procedure mentioned before, r can be solved as $r = 3.0$.

(c) Construct the two-point adaptive nonlinear approximation (TANA) using Eq. (3.3.47)

$$\begin{aligned} \tilde{g}(X) &= g(X_k) + \frac{1}{r}\sum_{i=1}^{2} x_{i,k}^{1-r}\frac{\partial g(X_k)}{\partial x_i}(x_i^r - x_{i,k}^r) \\ &= 573.8398 + \frac{1}{3}\cdot[6.6808^{-2} \times 133.8982 \times (x_1^3 - 6.6808^3) + \\ &\quad 6.6468^{-2} \times 132.5409 \times (x_2^3 - 6.6468^3)] \\ &= x_1^3 + x_2^3 - 18.0 \end{aligned}$$

(d) Transfer the above X-space approximate limit state function into the U-space function using Eq. (3.3.48)

$$\tilde{g}(U) = \tilde{g}(\sigma_{x_1}u_1 + \mu_{x_1}, \sigma_{x_2}u_2 + \mu_{x_2})$$

$$= (5u_1 + 10)^3 + (5u_2 + 9.9)^3 - 18.$$

(e) Find the most probable failure point X^* of the approximate safety index model given in Eq. (3.3.46) using DOT

After four iterations, the MPP point is found as

$$x_1^* = 2.0718, \qquad x_2^* = 2.0883$$

$$u_1^* = -1.5856, \qquad u_2^* = -1.5623$$

(f) Compute the safety index β_2

$$\begin{aligned} \beta_2 &= \sqrt{u_1^{*2} + u_2^{*2}} \\ &= \sqrt{(-1.5856)^2 + (-1.5623)^2} \\ &= 2.2260 \end{aligned}$$

(g) Convergence check

$$\varepsilon = \frac{|\beta_2 - \beta_1|}{\beta_1} = \frac{2.2260 - 0.9295}{0.9295} = 1.3948$$

Since $\varepsilon > \varepsilon_r(0.001)$, continue the process.

(3) Iteration 3:

(a) Compute the limit state function value at X^*

$$\begin{aligned} g(X^*) &= g(x_1^*, x_2^*) = x_1^{*3} + x_2^{*3} - 18 \\ &= 2.0718^3 + 2.0883^3 - 18 \\ &= -0.1276 \times 10^{-5} \end{aligned}$$

(b) Compute approximate gradients using the approximate limit state function

$$\frac{\partial \tilde{g}}{\partial x_1}|_\mu = 3x_1^{*2} = 3 \times 2.0718^2 = 12.8769$$

$$\frac{\partial \tilde{g}}{\partial x_2}|_\mu = 3x_2^{*2} = 3 \times 2.0883^2 = 13.0832$$

(c) Compute approximate safety index $\tilde{\beta}$ using the HL-RF method (Eq. 3.3.25) and the direction cosine α_i

$$\begin{aligned}
\tilde{\beta}_3 &= \frac{g(X^*) - \frac{\partial \tilde{g}(X^*)}{\partial x_1}\sigma_{x_1}u_{x_1}^* - \frac{\partial \tilde{g}(X^*)}{\partial x_2}\sigma_{x_2}u_{x_2}^*}{\sqrt{(\frac{\partial \tilde{g}(X^*)}{\partial x_1}\sigma_{x_1})^2 + (\frac{\partial \tilde{g}(X^*)}{\partial x_2}\sigma_{x_2})^2}} \\
&= \frac{-0.1276 \times 10^{-5} - 12.8769 \times 5 \times -1.5856 - 13.0832 \times 5 \times -1.5623}{\sqrt{(12.8769 \times 5)^2 + (13.0832 \times 5)^2}} \\
&= 2.2258
\end{aligned}$$

(d) Approximate convergence check

$$\varepsilon = \frac{|\tilde{\beta}_3 - \beta_2|}{\beta_2} = \frac{|2.2258 - 2.2260|}{2.2260} = 0.00009$$

Since $\varepsilon < \varepsilon_r(0.001)$, stop the process. The final safety index is 2.2258. Compared with the result of Example 3.3b ($\beta = 1.1657$), the safety index algorithm using TANA is much more efficient for this example. It only needs 3 g-function and 2 gradient calculations to reach the convergent point. Since the g-function value is very small, the final MPP is on the limit state surface.

In the above algorithm, the nonlinear approximation of the limit state function is performed in the original space of the random variables, hence the nonlinear approximation only represents the nonlinearity of the performance function in X-space and does not include the nonlinearity of the distribution transformation. In general, the nonlinearity of the transformed performance function $g(U)$ depends not only on the nonlinearity of the performance function in original space but also on the distribution transformation. Even a linear performance function in X-space may result in a nonlinear performance function in U-space because of the transformation. Therefore,

directly approximating the transformed performance function $g(U)$ of U-space instead of g(X) in X-space may be closer to the exact function for non-normal distributions. However, for U-space approximation, the intervening variables (Eq. 2.4.1) may become impractical because the standard variables in U-space are usually negative.

In order to construct an adaptive nonlinear approximation in U-space, improved intervening variables are denoted as $Y = (y_1, y_2, ...y_n)^T$.

$$y_i = (u_i + \frac{\mu_{x_i}}{\sigma_{x_i}} s)^r \qquad i = 1, 2, ..., n \tag{3.3.50}$$

where u_i $(i = 1, 2, ..., n)$ is the standard normal variable, μ_{x_i} $(i = 1, 2, ..., n)$ is the mean value, σ_{x_i} $(i = 1, 2, ..., n)$ is the standard deviation, r is the nonlinearity index, which mainly controls the functional nonlinearity, and s is the coefficient of coordinates shift which preserves the positiveness of normal coordinates and gives an additional degree of freedom for improving the accuracy. Both parameters, r and s can be selected appropriately by using the following feedback formula based on two analyses information at Y_{k-1} and Y_k (Y_k is the current point and Y_{k-1} is the previous point).

$$g(Y_{k-1}) - \{g(Y_k) + \frac{1}{r}\sum_i^n y_{i,k}^{(1-r)} \frac{\partial g(Y_k)}{\partial u_i}[(u_{i,k-1} + \frac{\bar{x}_i}{\sigma_i} s)^r - (u_{i,k} + \frac{\bar{x}_i}{\sigma_i} s)^r]\} = 0 \tag{3.3.51}$$

This equation has two unknown parameters r and s, which can be optimally determined by minimizing the error ε. In theory, the nonlinearity index r can be any positive or negative real number (not equal to 0), and the coefficient of coordinates shift s can be selected as any positive number (for simplicity it is selected as positive integer in this work). However, the nonlinearity index r increases fast when s increases, which may result in large r if s has a larger value for some problems. Therefore, in the algorithm implementation, r is restricted from -20 to 20, and the shift s is restricted from 1 to 15 for non-normal distributions (s is selected as 1 for normal distributions). The iteration for calculating r and s starts from $r = 1$ and $s = 1$. When r is increased or decreased a step length (0.1), s is changed from 1 to 15, and a combination of r and s with the smallest error ε is obtained. If this error is smaller than the initial error (e.g.

corresponding to $r = 1$ and $s = 1$), the above iteration is repeated until the allowable error $\varepsilon = 0.001$ or limitation of r is reached, and the optimum combination of r and s is determined. Otherwise, the step length of r is decreased by a half and the above iteration process is repeated until the final r and s are obtained.

Based on the above intervening variables given in Eq. (3.3.50), the nonlinear approximation in U-space can be obtained from the following equation

$$\tilde{g}(U) = g(Y_k) + \frac{1}{r}\sum_{i}^{n} y_{i,k}^{(1-r)} \frac{\partial g(Y_k)}{\partial u_i}[(u_i + \frac{\bar{x}_i}{\sigma_i}s)^r - (u_{i,k} + \frac{\bar{x}_i}{\sigma_i}s)^r] \tag{3.3.52}$$

The main steps of this U-space safety index algorithm are summarized as follows:

(1) In the first iteration, construct a linear approximation to the original performance function by using a first-order Taylor's series expansion about the mean values of the random variables, μ;

(2) Transfer the original random variables in X-space to the standard normal variables in U-space by using the approximate distribution transformation of Eq.(3.3.39);

(3) Compute the most probable failure point U_k and safety index β_k using the RF method in U-space;

(4) Establish the feedback formula of U-space Eq. (3.3.51) by using the intervening variables of Eq. (3.3.50) and the first-order Taylor's series expansion based on the information of the current point U_k and the previous point U_{k-1};

(5) Determine the optimum combination of the nonlinearity index r and the coefficient s by minimizing the error ε of Eq. (3.3.51);

(6) Construct the nonlinear approximation of the performance function (3.3.52) based on the intervening variables Eq. (3.3.50);

(7) Find the most probable failure point U_k of the nonlinear approximate function $\tilde{g}(U)$ and the safety index β_k using the HL-RF or DOT;

(8) Check the convergence

$$\varepsilon_\beta = |\frac{\beta_k - \beta_{k-1}}{\beta_{k-1}}|$$

If the convergence is not satisfied, go to step (9);

(9) Compute the exact value of the g-function at U_k and using this value and the approximate gradients obtained from $\tilde{g}(U)$ solve for the safety index β_a;

(10) Check the convergence

$$\varepsilon_a = |\frac{\beta_a - \beta_k}{\beta_k}|$$

If the convergence is not satisfied, compute the exact gradients,and go to step (4) and repeat the process until convergence is reached.

A significant reduction in computer effort results from the use of the approximate function in step (7). Exact function evaluation is avoided and $\tilde{g}(U)$ given in Eq. (3.3.52) is used to solve the function value and gradients. This method is particularly suitable for problems with highly nonlinear and implicit performance functions needing large scale finite element models for structural analysis.

3.3.3.4 Approximate Limit State Function Using Improved Two-point Adaptive Nonlinear Approximation

The difference between this algorithm and the above algorithm is that the limit state function is approximated by the TANA2 approximation given in Eq. (2.4.9) since TANA2 provides better accuracy in some complex cases. The main steps of the X-space algorithm using TANA2 are summarized as follows:

1). In the first iteration, compute the mean and standard deviation of the equivalent normal distribution at the mean value point for nonnormal distribution variables. Construct a linear approximation of Eq. (3.3.23) by using the first-order Taylor's series expansion at an initial point (if the initial point is selected as the mean value point, μ, the linear approximation is expanded at μ), and compute the limit state function value and gradients at the initial point;

2). Compute the initial safety index β_1 using the HL-RF method and its direction cosine α_i (if the initial point is the mean value point, the mean value method is used);

3). Compute the new design point using Eq. (3.3.28), X_k;

4). Compute the mean and standard deviation of the equivalent normal distribution at X_k for nonnormal distribution variables. Calculate the limit state function value and gradients at the new design point, X_k;

5). Determine the nonlinearity index p_k $(k = 1, 2, ..., N)$ by solving Equations (2.4.10) and (2.4.11) based on the information of the current and previous points (when k equals to 2, previous design point is the mean value $\bar{X}$);

6). Obtain the adaptive nonlinear approximation of Eq.(2.4.9);

7). Transform the X-space approximate limit state function into the U-space function using Eq. (3.3.48);

8). Find the most probable failure point X_{k+1} of the approximate safety index model given in Eq. (3.3.46) using the HL-RF method or DOT and compute the safety index β_{k+1};

9). Check the convergence

$$\varepsilon = |\frac{\beta_{k+1} - \beta_k}{\beta_k}|$$

10). Stop the process if ε satisfies the required convergence tolerance (0.001), otherwise, Continue;

11). Compute the exact limit state function value and approximate gradients at X_{k+1}, and estimate the approximate safety index $\tilde{\beta}_{k+1}$ using the HL-RF method;

12). Approximate β convergence check

$$\varepsilon = \frac{|\tilde{\beta}_{k+1} - \beta_k|}{\beta_k}$$

13). Stop the process if ε satisfies the required convergence tolerance (0.001), otherwise, Continue;

14). Compute the exact gradients of the limit state function at X_{k+1} and go to step 5); repeat the process until β converges.

In step (8), the safety index β is iteratively computed for the explicit approximate function $\tilde{g}(X)$. Any iterative algorithm can be used for finding the MPP. The computation of the exact performance function $\tilde{g}(X)$ is not required; therefore, the computer time is greatly reduced for problems involving complex and implicit performance functions, particularly with finite element models.

Example 3.9

The performance function is

$$g(x_1, x_2) = x_1 x_2 - 1400$$

in which x_1 and x_2 are the random variables with lognormal distributions. The mean values and standard deviations of two variables are: $\mu_{x_1} = 40.0$, $\mu_{x_2} = 50.0$, $\sigma_{x_1} = 5.0$, $\sigma_{x_2} = 2.5$. The safety index algorithm using TANA2 is used to solve the safety index β.

(1) Compute the mean values and standard deviations of a normally distributed variables y_1 and y_2 ($y_1 = lnx_1$, $y_2 = lnx_2$) using Eqs. (1.53) and (1.54),

$$\begin{aligned} \sigma_{y_1} &= \sqrt{ln[(\frac{\sigma_{x_1}}{\mu_{x_1}})^2 + 1]} \\ &= \sqrt{ln[(\frac{5.0}{40.0})^2 + 1]} \\ &= 0.1245 \end{aligned}$$

$$\begin{aligned} \sigma_{y_2} &= \sqrt{ln[(\frac{\sigma_{x_2}}{\mu_{x_2}})^2 + 1]} \\ &= \sqrt{ln[(\frac{2.5}{50.0})^2 + 1]} \\ &= 4.9969 \times 10^{-2} \end{aligned}$$

$$\begin{aligned}\mu_{y_1} &= ln\mu_{x_1} - \frac{1}{2}\sigma_{y_1}^2 \\ &= ln40 - \frac{1}{2} \times 0.1245^2 \\ &= 3.6811\end{aligned}$$

$$\begin{aligned}\mu_{y_2} &= ln\mu_{x_2} - \frac{1}{2}\sigma_{y_2}^2 \\ &= ln50 - \frac{1}{2} \times (4.9969 \times 10^{-2})^2 \\ &= 3.9108\end{aligned}$$

(2) Iteration 1:

(a) Compute the mean values and standard deviations of the equivalent normal distributions for x_1 and x_2:

First, assuming the design point, $X^* = \{x_1^*, x_2^*\}^T$, as the mean value point, the coordinates of the initial design point are

$$x_1^* = \mu_{x_1} = 2 \times 40.0, \qquad x_2^* = \mu_{x_2} = 50.0$$

The density function values at x_1^* and x_2^* are

$$\begin{aligned}f_{x_1}(x_1^*) &= \frac{1}{\sqrt{2\pi}x_1^*\sigma_{y_1}} exp[-\frac{1}{2}(\frac{lnx_1^* - \mu_{y_1}}{\sigma_{y_1}})^2] \\ &= \frac{1}{\sqrt{2\pi} \times 40 \times 0.1245} exp[-\frac{1}{2}(\frac{ln40 - 3.6811}{0.1245})^2] \\ &= 7.9944^{-2}\end{aligned}$$

$$f_{x_2}(x_2^*) = \frac{1}{\sqrt{2\pi}x_2^*\sigma_{y_2}} exp[-\frac{1}{2}(\frac{lnx_2^* - \mu_{y_2}}{\sigma_{y_2}})^2]$$

$$= \frac{1}{\sqrt{2\pi} \times 50 \times 4.9969 \times 10^{-2}} exp[-\frac{1}{2}(\frac{ln50 - 3.9108}{4.9969 \times 10^{-2}})^2]$$
$$= 0.1596$$

$$\phi(\Phi^{-1}[F_{x_1}(x_1^*)]) = \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx_1^* - \mu_{y_1}}{\sigma_{y_1}})^2]$$
$$= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln40 - 3.6811}{0.1245})^2]$$
$$= 0.3982$$

$$\phi(\Phi^{-1}[F_{x_2}(x_2^*)]) = \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx_2^* - \mu_{y_2}}{\sigma_{y_2}})^2]$$
$$= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln50 - 3.9108}{4.9969 \times 10^{-2}})^2]$$
$$= 0.3988$$

Therefore the standard deviation and mean value of the equivalent normal variable at P^* from Eqs. (3.3.44a) and (3.3.44b) are

$$\sigma_{x_1'} = \frac{\phi(\Phi^{-1}[F_{x_1}(x_1^*)])}{f_{x_1}(x_1^*)}$$
$$= \frac{0.3982}{7.9944 \times 10^{-2}}$$
$$= 4.9806$$

$$\sigma_{x_2'} = \frac{\phi(\Phi^{-1}[F_{x_2}(x_2^*)])}{f_{x_2}(x_2^*)}$$
$$= \frac{0.3988}{0.1596}$$
$$= 2.4984$$

$$\begin{aligned}\mu_{x_1'} &= x_1^* - \Phi^{-1}[F_{x_1}(x_1^*)]\sigma_{x_1'} \\ &= 40 - 6.2258^{-2} \times 4.9806 \\ &= 39.6899\end{aligned}$$

$$\begin{aligned}\mu_{x_2'} &= x_2^* - \Phi^{-1}[F_{x_2}(x_2^*)]\sigma_{x_2'} \\ &= 50 - 2.4984 \times 10^{-2} \times 2.4984 \\ &= 49.9376\end{aligned}$$

(b) Set the mean value point as an initial design point and required β convergence tolerance as $\varepsilon_r = 0.001$. Compute the limit state function value and gradients at the mean value point

$$\begin{aligned}g(X^*) &= g(\mu_{x_1}, \mu_{x_2}) = \mu_{x_1}\mu_{x_2} - 1400 \\ &= 40 \times 50 - 1400 \\ &= 600.0\end{aligned}$$

$$\frac{\partial g}{\partial x_1}|_\mu = \mu_{x_2} = 50$$
$$\frac{\partial g}{\partial x_2}|_\mu = \mu_{x_1} = 40$$

(c) Compute the initial β using the mean value method and its direction cosine α_i

$$\begin{aligned}\beta_1 &= \frac{\mu_{\tilde{g}}}{\sigma_{\tilde{g}}} \\ &= \frac{g(X^*)}{\sqrt{(\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_1}\sigma_{x_1'})^2 + (\frac{\partial g(\mu_{x_1},\mu_{x_2})}{\partial x_2}\sigma_{x_2'})^2}}\end{aligned}$$

$$= \frac{600}{\sqrt{(50 \times 4.9806)^2 + (40 \times 2.4984)^2}}$$
$$= 2.1689$$

$$\alpha_1 = -\frac{\frac{\partial g}{\partial x_1}|_\mu \sigma_{x_1'}}{\sqrt{(\frac{\partial g(\mu_{x_1}, \mu_{x_2})}{\partial x_1}\sigma_{x_1'})^2 + (\frac{\partial g(\mu_{x_1}, \mu_{x_2})}{\partial x_2}\sigma_{x_2'})^2}}$$
$$= -\frac{50 \times 4.9806}{\sqrt{(50 \times 4.9806)^2 + (40 \times 2.4984)^2}}$$
$$= -0.9281$$

$$\alpha_2 = -\frac{\frac{\partial g}{\partial x_2}|_\mu \sigma_{x_2'}}{\sqrt{(\frac{\partial g(\mu_{x_1}, \mu_{x_2})}{\partial x_1}\sigma_{x_1'})^2 + (\frac{\partial g(\mu_{x_1}, \mu_{x_2})}{\partial x_2}\sigma_{x_2'})^2}}$$
$$= -\frac{40 \times 2.4984}{\sqrt{(50 \times 4.9806)^2 + (40 \times 2.4984)^2}}$$
$$= -0.3724$$

(d) Compute a new design point X^* from Eq. (3.3.28)

$$x_1^* = \mu_{x_1'} + \beta_1 \sigma_{x_1'} \alpha_1$$
$$= 39.6899 + 2.1689 \times 4.9806 \times (-0.9281)$$
$$= 29.6645$$

$$x_2^* = \mu_{x_2'} + \beta_1 \sigma_{x_2'} \alpha_2$$
$$= 49.9376 + 2.1689 \times 2.4984 \times (-0.3724)$$
$$= 47.9194$$

$$u_1^* = \frac{x_1^* - \mu_{x_1'}}{\sigma_{x_1'}} = -2.0129$$

$$u_2^* = \frac{x_2^* - \mu_{x_2'}}{\sigma_{x_2'}} = -0.8078$$

(3) Iteration 2:

(a) Compute the mean values and standard deviations of the equivalent normal distributions for x_1^* and x_2^*:

The density function values at x_1^* and x_2^* are

$$\begin{aligned} f_{x_1}(x_1^*) &= \frac{1}{\sqrt{2\pi}x_1^*\sigma_{y_1}} exp[-\frac{1}{2}(\frac{lnx_1^* - \mu_{y_1}}{\sigma_{y_1}})^2] \\ &= \frac{1}{\sqrt{2\pi} \times 29.6645 \times 0.1245} exp[-\frac{1}{2}(\frac{ln29.6645 - 3.6811}{0.1245})^2] \\ &= 7.0144^{-3} \end{aligned}$$

$$\begin{aligned} f_{x_2}(x_2^*) &= \frac{1}{\sqrt{2\pi}x_2^*\sigma_{y_2}} exp[-\frac{1}{2}(\frac{lnx_2^* - \mu_{y_2}}{\sigma_{y_2}})^2] \\ &= \frac{1}{\sqrt{2\pi} \times 47.9194 \times 4.9969 \times 10^{-2}} exp[-\frac{1}{2}(\frac{ln47.9194 - 3.9108}{4.9969 \times 10^{-2}})^2] \\ &= 0.1185 \end{aligned}$$

$$\begin{aligned} \phi(\Phi^{-1}[F_{x_1}(x_1^*)]) &= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx_1^* - \mu_{y_1}}{\sigma_{y_1}})^2] \\ &= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln29.6645 - 3.6811}{0.1245})^2] \\ &= 2.5909^{-2} \end{aligned}$$

$$\phi(\Phi^{-1}[F_{x_2}(x_2^*)]) = \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx_2^* - \mu_{y_2}}{\sigma_{y_2}})^2]$$

$$= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln47.9194 - 3.9108}{4.9969 \times 10^{-2}})^2]$$
$$= 0.2837$$

Therefore the standard deviation and mean value of the equivalent normal variable at P^* from Eqs. (3.3.44a) and (3.3.44b) are

$$\sigma_{x_1'} = \frac{\phi(\Phi^{-1}[F_{x_1}(x_1^*)])}{f_{x_1}(x_1^*)}$$
$$= \frac{2.5909^{-2}}{7.0144^{-3}}$$
$$= 3.6937$$

$$\sigma_{x_2'} = \frac{\phi(\Phi^{-1}[F_{x_2}(x_2^*)])}{f_{x_2}(x_2^*)}$$
$$= \frac{0.2837}{0.1185}$$
$$= 2.3945$$

$$\mu_{x_1'} = x_1^* - \Phi^{-1}[F_{x_1}(x_1^*)]\sigma_{x_1'}$$
$$= 29.6645 - (-2.3384) \times 3.6937$$
$$= 38.3021$$

$$\mu_{x_2'} = x_2^* - \Phi^{-1}[F_{x_2}(x_2^*)]\sigma_{x_2'}$$
$$= 47.9194 - (-0.8256) \times 2.3945$$
$$= 49.8963$$

(b) Compute the limit state function value and gradients at X^*

$$
\begin{aligned}
g(X^*) &= g(x_1^*, x_2^*) = x_1^* x_2^* - 1400 \\
&= 29.6645 \times 47.9194 - 1400 \\
&= 21.5041
\end{aligned}
$$

$$
\frac{\partial g}{\partial x_1}|_{X^*} = x_2^* = 47.9194
$$
$$
\frac{\partial g}{\partial x_2}|_{X^*} = x_1^* = 29.6645
$$

(c) Compute the nonlinearity indices p_1 and p_2 based on the function values and gradients of the two points $\mu(40, 50)$ and $X^*(29.6645, 47.9194)$ using Eqs. (2.4.10) and (2.4.11), that is

$$
\frac{\partial g(\mu)}{\partial x_1} = (\frac{\mu_1}{x_1^*})^{p_1 - 1} \frac{\partial g(X^*)}{\partial x_1} + \varepsilon_2 (\mu_1^{p_1} - (x_1^*)^{p_1}) \mu_1^{p_1 - 1} p_1
$$

$$
50 = (\frac{40}{29.6645})^{p_1 - 1} 47.9194 + \varepsilon_2 (40^{p_1} - (47.9194^*)^{p_1}) 40^{p_1 - 1} p_1
$$

$$
\frac{\partial g(\mu)}{\partial x_2} = (\frac{\mu_2}{x_2^*})^{p_2 - 1} \frac{\partial g(X^*)}{\partial x_2} + \varepsilon_2 (\mu_2^{p_2} - (x_2^*)^{p_2}) \mu_2^{p_2 - 1} p_2
$$

$$
40 = (\frac{50}{47.9194})^{p_2 - 1} 29.6645 + \varepsilon_2 (50^{p_2} - (47.9194^*)^{p_2}) 50^{p_2 - 1} p_2
$$

$$
g(\mu) = g(X^*) + \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \frac{(x_i^*)^{1 - p_i}}{p_i} (\mu_i^{p_i} - (x_i^*)^{p_i}) + \frac{1}{2} \varepsilon_2 \sum_{i=1}^{2} (\mu_i^{p_i} - (x_i^*)^{p_i})^2
$$

$$
\begin{aligned}
600 &= 21.5041 + 47.9194 \times \frac{29.6645^{1 - p_1}}{p_1} (40^{p_1} - 29.6645^{p_1}) \\
&\quad + 29.6645 \times \frac{47.9194^{1 - p_2}}{p_2} (50^{p_1} - 47.9194^{p_2} \\
&\quad + \frac{1}{2} \varepsilon_2 [(\mu_1^{p_1} - (x_1^*)^{p_1})^2 + (\mu_2^{p_2} - (x_2^*)^{p_2})^2]
\end{aligned}
$$

Based on the above three equations, p_i and ε_2 can be solved using the adaptive search procedure.

$$p_1 = 1.0375, \qquad p_2 = 1.4125, \qquad \varepsilon_2 = 0.1$$

(d) Construct the improved two-point adaptive nonlinear approximation (TANA2) using Eq. (2.4.9)

$$\begin{aligned}
\tilde{g}(X) &= g(X^*) + \sum_{i=1}^{2} \frac{\partial g(X^*)}{\partial x_i} \frac{(x_i^*)^{1-p_i}}{p_i}(x_i^{p_i} - (x_i^*)^{p_i}) + \frac{1}{2}\varepsilon_2 \sum_{i=1}^{2}(x_i^{p_i} - (x_i^*)^{p_i})^2 \\
&= 21.5041 + 47.9194 \times \frac{29.6645^{1-1.0375}}{1.0375}(x_1^{1.0375} - 29.6645^{1.0375}) \\
&\quad +29.6645 \times \frac{47.9194^{1-1.4125}}{1.4125}(x_2^{1.4125} - 47.9194^{1.4125}) \\
&\quad +\frac{0.1}{2}[(x_1^{1.0375} - 29.6645^{1.0375})^2 + (x_2^{1.4125} - 47.9194^{1.4125})^2] \\
&= 21.5041 + 40.6738(x_1^{1.0375} - 29.6645^{1.0375}) + 4.2564(x_2^{1.4125} - 47.9194^{1.4125}) \\
&\quad +\frac{0.1}{2}[(x_1^{1.0375} - 29.6645^{1.0375})^2 + (x_2^{1.4125} - 47.9194^{1.4125})^2]
\end{aligned}$$

(e) Transfer the above X-space approximate limit state function into the U-space function using Eq. (3.3.48)

$$\begin{aligned}
\tilde{g}(U) &= \tilde{g}(\sigma_{x_1'} u_1 + \mu_{x_1'}, \sigma_{x_2'} u_2 + \mu_{x_2'}) \\
&= 21.5041 + 40.6738[(3.6937\mu_1 + 38.3021)^{1.0375} - 29.6645^{1.0375}] \\
&\quad +4.2564[(2.3945\mu_2 + 49.8963)^{1.4125} - 47.9194^{1.4125}] \\
&\quad +\frac{0.1}{2}[((3.6937\mu_1 + 38.3021)^{1.0375} - 29.6645^{1.0375})^2 \\
&\quad +((2.3945\mu_2 + 49.8963)^{1.4125} - 47.9194^{1.4125})^2]
\end{aligned}$$

(f) Find the most probable failure point X^* of the approximate safety index model given in Eq. (3.3.46) using DOT

After two iterations, the MPP point is found as

$$x_1^* = 29.3517, \qquad x_2^* = 47.6961$$

$$u_1^* = -2.4236, \qquad u_2^* = -0.9191$$

At each iteration, the mean value and standard deviation of the equivalent normal distributions at the new design point X^* need to be calculated.

(g) Compute the safety index β_2

$$\begin{aligned} \beta_2 &= \sqrt{u_1^{*2} + u_2^{*2}} \\ &= \sqrt{(-2.4236)^2 + (-0.9191)^2} \\ &= 2.5920 \end{aligned}$$

(h) Convergence check

$$\varepsilon = \frac{|\beta_2 - \beta_1|}{\beta_1} = \frac{2.5920 - 2.1689}{2.1689} = 0.1951$$

Since $\varepsilon > \varepsilon_r(0.001)$, continue the process.

(4) Iteration 3:

(a) Compute the mean values and standard deviations of the equivalent normal distributions for x_1^* and x_2^*:

The density function values at x_1^* and x_2^* are

$$\begin{aligned} f_{x_1}(x_1^*) &= \frac{1}{\sqrt{2\pi}x_1^*\sigma_{y_1}} exp[-\frac{1}{2}(\frac{lnx_1^* - \mu_{y_1}}{\sigma_{y_1}})^2] \\ &= \frac{1}{\sqrt{2\pi} \times 29.3517 \times 0.1245} exp[-\frac{1}{2}(\frac{ln29.3517 - 3.6811}{0.1245})^2] \\ &= 5.7886 \times 10^{-3} \end{aligned}$$

$$
\begin{aligned}
f_{x_2}(x_2^*) &= \frac{1}{\sqrt{2\pi}x_2^*\sigma_{y_2}} exp[-\frac{1}{2}(\frac{lnx_2^* - \mu_{y_2}}{\sigma_{y_2}})^2] \\
&= \frac{1}{\sqrt{2\pi} \times 47.6961 \times 4.9969 \times 10^{-2}} exp[-\frac{1}{2}(\frac{ln47.6961 - 3.9108}{4.9969 \times 10^{-2}})^2] \\
&= 0.1097
\end{aligned}
$$

$$
\begin{aligned}
\phi(\Phi^{-1}[F_{x_1}(x_1^*)]) &= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx_1^* - \mu_{y_1}}{\sigma_{y_1}})^2] \\
&= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln29.3517 - 3.6811}{0.1245})^2] \\
&= 2.1156^{-2}
\end{aligned}
$$

$$
\begin{aligned}
\phi(\Phi^{-1}[F_{x_2}(x_2^*)]) &= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{lnx_2^* - \mu_{y_2}}{\sigma_{y_2}})^2] \\
&= \frac{1}{\sqrt{2\pi}} exp[-\frac{1}{2}(\frac{ln47.6961 - 3.9108}{4.9969 \times 10^{-2}})^2] \\
&= 0.2615
\end{aligned}
$$

Therefore the standard deviation and mean value of the equivalent normal variable at P^* from Eqs. (3.3.44a) and (3.3.44b) are

$$
\begin{aligned}
\sigma_{x_1'} &= \frac{\phi(\Phi^{-1}[F_{x_1}(x_1^*)])}{f_{x_1}(x_1^*)} \\
&= \frac{2.1156^{-2}}{5.7886^{-3}} \\
&= 3.6548
\end{aligned}
$$

$$
\begin{aligned}
\sigma_{x_2'} &= \frac{\phi(\Phi^{-1}[F_{x_2}(x_2^*)])}{f_{x_2}(x_2^*)} \\
&= \frac{0.2615}{0.1097} \\
&= 2.3833
\end{aligned}
$$

$$\begin{aligned}\mu_{x_1'} &= x_1^* - \Phi^{-1}[F_{x_1}(x_1^*)]\sigma_{x_1'} \\ &= 29.3517 - (-2.4236) \times 3.6548 \\ &= 38.2094\end{aligned}$$

$$\begin{aligned}\mu_{x_2'} &= x_2^* - \Phi^{-1}[F_{x_2}(x_2^*)]\sigma_{x_2'} \\ &= 47.6961 - (-0.9191) \times 2.3833 \\ &= 49.8865\end{aligned}$$

(b) Compute the limit state function value at X^*

$$\begin{aligned}g(X^*) &= g(x_1^*, x_2^*) = x_1^* x_2^* - 1400 \\ &= 29.3517 \times 47.6961 - 1400 \\ &= -0.0359\end{aligned}$$

(c) Compute approximate gradients using the approximate limit state function

$$\frac{\partial \tilde{g}}{\partial x_1}|_\mu = 47.8569$$
$$\frac{\partial \tilde{g}}{\partial x_2}|_\mu = 28.5261$$

(d) Compute approximate safety index $\tilde{\beta}$ using the HL-RF method (Eq. 3.3.25) and the direction cosine α_i

$$\begin{aligned}\tilde{\beta}_3 &= \frac{g(X^*) - \frac{\partial \tilde{g}(X^*)}{\partial x_1}\sigma_{x_1'}u_{x_1}^* - \frac{\partial \tilde{g}(X^*)}{\partial x_2}\sigma_{x_2'}u_{x_2}^*}{\sqrt{(\frac{\partial \tilde{g}(X^*)}{\partial x_1}\sigma_{x_1'})^2 + (\frac{\partial \tilde{g}(X^*)}{\partial x_2}\sigma_{x_2'})^2}} \\ &= \frac{-0.0359 - 47.8569 \times 3.6548 \times -2.4236 - 28.5261 \times 2.3833 \times -0.9191}{\sqrt{(47.8569 \times 3.6548)^2 + (28.5261 \times 2.3833)^2}} \\ &= 2.5917\end{aligned}$$

(e) Approximate convergence check

$$\varepsilon = \frac{|\tilde{\beta}_3 - \beta_2}{\beta_2} = \frac{|2.5917 - 2.5920|}{2.5920} = 0.0001$$

Since $\varepsilon < \varepsilon_r(0.001)$, stop the process. The final safety index is 2.5917.

3.3.3.5 Algorithms Comparison

In this section, several examples are used to compare the efficiency of the algorithms introduced in this chapter. These examples include problems with highly nonlinear, explicit and implicit performance functions which required finite element analyses. Furthermore, the summary of the safety index analysis is given.

Example 3.10

The performance function is given in Example 3.1. Two cases are considered in this example. In case 1, the mean values of x_1 and x_2 are 10 and 9.9, respectively, and the standard deviations of both variables are 5.0 (the same as in Example 3.3b). The mean and standard deviation of the case 2 are 10 and 5.0, respectively (the same as in Example 3.1). The comparison of the results obtained from the above four algorithms and the mean value method is shown in Table 3.1.

For case 1, the above algorithms 3 and 4 need only 3 g-function calculations and 2 gradient calculations to find the convergent solution, while the HL-RF method and algorithm 2 fail to the correct MPP on the limit state surface after 23 and 22 iterations, respectively, due to oscillation. Fig. 3.13 shows the MPP and how the algorithms 3 and 4 converge in three steps. HL-RF method approaches the limit state surface in the first few iterations, but after the fifth step, it completely diverges and oscillates away from the solution. For case 2, all the methods converge with a different number of iterations (Fig. 3.14). The algorithms 3 and 4 also need 3 g-function calculations and 2 gradient calculations to find the convergent solution, while the HL-RF method and algorithm 2 need 7 iterations to converge.

Table 3.1 Comparison of Methods for Example 3.10

Case	Case 1				Case 2			
Method	HL-RF	Method 2	Method 3	Method 4	HL-RF	Method 2	Method 3	Method 4
Iter. 1	0.9295	0.9295	0.9295	0.9295	0.9343	0.9343	0.9343	0.9343
Iter. 2	1.5387	1.5387	2.2260(3.0*)	2.2260(3.0, 3.0**)	1.5468	1.5468	2.2401(3.0*)	2.2401(3.0, 3.0**)
Iter. 3	1.9222	1.9222	2.2258	2.2258	1.9327	1.9327	2.2401	2.2401
Iter. 4	2.1339	2.1342	—	—	2.1467	2.1467	—	—
Iter. 5	2.2001	2.2102	—	—	2.2279	2.2279	—	—
Iter. 6	1.8045	2.0522	—	—	2.2398	2.2398	—	—
Iter. 7	0.9429	1.2065	—	—	2.2401	2.2401	—	—
....								
Iter.22	1.1650	1.1613	—	—	—	—	—	—
Iter.23	1.1657	—	—	—	—	—	—	—
Final β	1.1657	1.1613	2.2258	2.2258	2.2401	2.2401	2.2401	2.2401
g-func. cal.	23	22	3	3	7	7	3	3
Grad. cal.	23	22	2	2	7	7	2	2

1. The values with an asterisk (*) represent the nonlinearity index r for TANA.
2. The values with an double asterisk (**) are the nonlinearity indices p_1 and p_2 for TANA2.

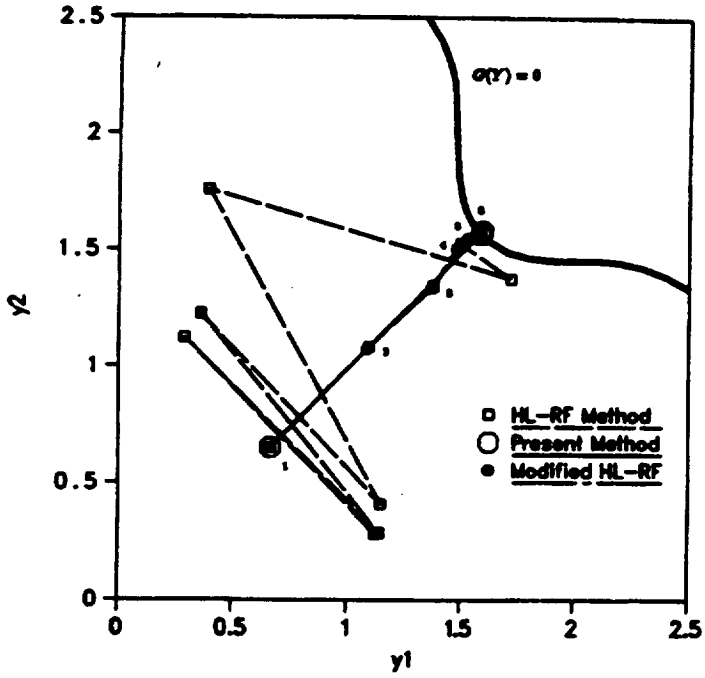


Fig.3.13 Iteration History of Example 3.10 (case 1)

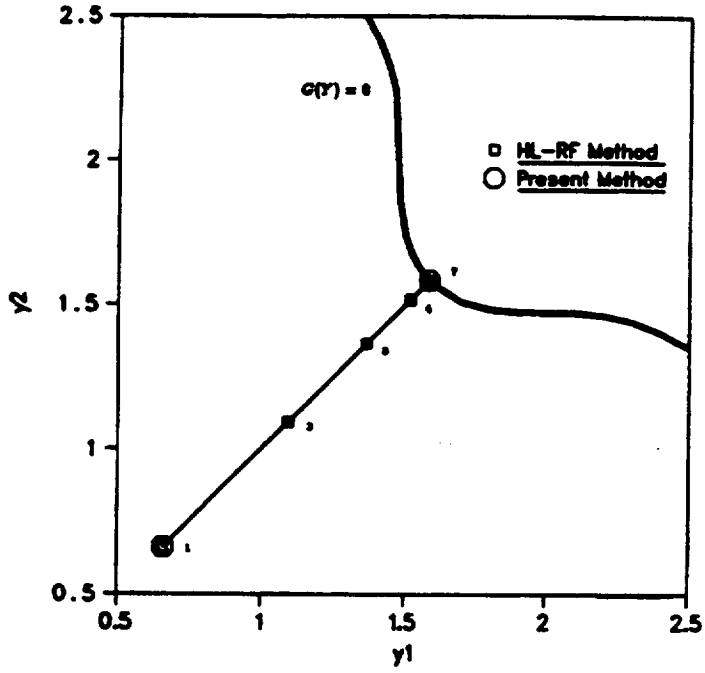


Fig.3.14 Iteration History of Example 3.10 (case 2)

Example 3.11

This example is taken from the NESSUS manual [12]. The performance function is

$$g(x_1, x_2) = x_1^2 + x_2 - 8.7577$$

in which x_1 and x_2 are the random variables with normal distributions (mean=10, Cov=0.3). The results comparison is listed in Table 3.2, which shows that the present method converges fast. Also, it shows the iteration history of the present method, and r is quite close to the nonlinearity of the real performance function.

Example 3.12

This example has a highly nonlinear performance function

$$g(x_1, x_2) = x_1^4 + 2x_2^4 - 20$$

in which x_1 and x_2 are the random variables with normal distributions (mean $\mu_{x_1} = \mu_{x_2} = 10.0$, standard deviation $\sigma_{x_1} = \sigma_{x_2} = 5.0$). The comparison of safety index results is shown in Table 3.3. In this example, the coefficient of coordinates shift s is selected as 1 and the nonlinearity index r is always equal to 4 in each step, this is the same as the nonlinearity of the performance function. The comparison results show that methods 3 and 4 are quite efficient in both X-space and U-space with only two iterations and an additional function calculation at the end of the iteration process. The HL-RF method and algorithm 2 did not converge even after 101 iterations. These two methods started from the mean values and approached the constraint surface in the first few iterations, but after the fifth iteration, they completely diverged from the constraint surface. After several iterations they oscillated between $\beta = 0.9267$ and $\beta = 0.9862$.

Example 3.13 313-member frame

This example has an implicit performance function needing finite element analysis. The frame structure shown in Fig. 2.3 has an I section modelled with 313 elements. Young's

Table 3.2 Comparison of Methods for Example 3.11

Method	HL-RF	Method 2	Method 3	Method 4
Iter. 1	1.6853	1.6853	1.6853	1.6853
Iter. 2	2.5372	2.5372	3.1076 (2.0*)	3.0588 (2.0, 1.0**)
Iter. 3	2.9521	2.9521	3.0584	3.0588
			3.0588 (1.975*)	—
Iter. 4	3.0583	3.0583	—	—
Iter. 5	3.0585	3.0585	—	—
Final β	3.0585	3.0585	3.0588	3.0588
g-func. calculations.	5	5	3	3
Gradients calculations	5	5	3	2

1. The values with an asterisk ($*$) represent the nonlinearity index r for TANA.
2. The values with an double asterisk ($**$) are the nonlinearity indices p_1 and p_2 for TANA2.

Table 3.3 Comparison of Methods for Example 3.12

Method	HL-RF	Method 2	Method 3	Method 4
Iter. 1	0.6704	0.6704	0.6704	0.6704
Iter. 2	1.1900	1.1900	2.3655 (4.0*)	2.3655(4.0, 4.0**)
Iter. 3	1.5685	1.5686	2.3655	2.3655
Iter. 4	1.6000	1.6016	—	—
...				
Iter. 101	0.9267	0.9267	—	—
Final β	0.9267	1.6016	2.3655	2.3655
g-func. calculations.	101(not conver.)	101(not conver.)	3	3
Gradients calculations	101(not conver.)	101(not conver.)	2	2

1. The values with an asterisk ($*$) represent the nonlinearity index r for TANA.
2. The values with an double asterisk ($**$) are the nonlinearity indices p_1 and p_2 for TANA2.

modulus E and the cross-sectional areas of all members are selected as the random variables with normal distributions, and the total number of random variables is 314. The area moment of inertia I_z is expressed as explicit nonlinear functions of A in the form $I_z = 0.2072A$. The vertical loads at nodes 15, 16, 88, 89 are -26, -30, -18, -20 *kips*, respectively. The horizontal loads at nodes 6, 11, 17 through 65 by 3, 68 through 82 by 7, and 90 through 175 by 5 are 4 *kips*. The horizontal load at node 1 is 2 *kips*. After the first analysis, the most critical constraint of the total 358 displacement constraints for this load case, i.e. the vertical displacement d at point 16 is taken as the limit state function, which is an implicit function of random variables and is written as

$$g(X) = 1. - d/d_{lim}$$

where d_{lim} is the displacement limit of 4.0 in. The mean value of Young's modulus E is 2.9×10^7 psi, with coefficient of variation (Cov) 0.08; all the element areas have a mean of 28.0 in^2, with Cov 0.08. The coefficient of coordinates shift s is selected as 1. The comparison of results presented in Table 3.4 shows that algorithm 3 in both X-space and U-space are very efficient for this complex implicit problem; only four iterations were needed to converge even though HL-RF method needed 45 iterations to converge to a local solution. The n_g count in Table 3.4 assumes finite difference calculation of gradients with respect to 314 random variables.

3.4 Summary

In this chapter, some basic concepts on the safety index and MPP calculations were introduced. The details of the Mean Value, HL, HL-RF and developed safety index algorithms using approximations were given. The numerical results showed that the mean value and HL/HL-RF method work well for the linear problems, however the HL/HL-RF may not converge even though many iterations are reached for highly nonlinear problems. The safety index algorithms using two-point adaptive nonlinear approximations are more efficient and stable

Table 3.4 Results Comparison for 313 Member Frame

Method	HL-RF		Algorithm 3 U-Space			Algorithm 3 X-Space		
Iter. No.	β	n_g	β	r	n_g	β	r	n_g
1	1.0513	315	1.0513	–	315	1.0513	–	315
2	2.0343	315	3.4146	-2.5	315	3.4147	-2.5	315
3	2.9496	315	3.8542	-0.5	315	3.8549	-0.5	315
4	3.7982	315	3.8521	1.0	315	3.8517	1.0	315
5	4.5854	315						
6	5.3146	315						
...	...	...						
45	12.540	315						
Total n_g		14175			1260			1260
Final Value of g-function	0.000442 Local Solution		0.000265 Converged			0.000466 Converged		

than HL/HL-RF for the highly nonlinear problems. In particular, the method with TANA2 works the best compared to other methods in most cases.

*

References

[1] Cornell, C. A., "A Probability-based Structural Code", Journal of the American Concrete Institute, Vol. 66, No. 12, 1969, pp. 974-985.

[2] Madsen, H. O., Krenk, S. and Lind, N. C., Methods of Structural Safety, Prentice-Hall, Englewood Cliffs, New Jersey, 1986.

[3] Liu, Pei-ling and Der Kiureghian, A., "Optimization Algorithms for Structural Reliability Analysis", Report No. UCB/SESM-86/09, Department of Civil Engineering, University of California, Berkeley, July, 1986.

[4] Hasofer, A. M. and Lind, N. C., "Exact and Invariant Second-moment Code Format", Journal of the Engineering Mechanics Division, ASCE, Vol. 100, No. 1, 1974, pp. 111-121.

[5] Rosenblatt, M., "Remarks on a Multivariate Transformation", The Annals of Mathematical Statistics, Vol. 23, No. 3, 1952, pp. 470-472.

[6] Hohenbichler, M. and Rackwitz, R., "Non-normal Dependent Vectors in Structural Safety", Journal of the Engineering Mechanics Division, ASCE, Vol. 107, No. EM6, Dec. 1981, pp. 1227-1238.

[7] Melchers, R. E., Structural Reliability Analysis and Prediction, Ellis Horwood Limited, UK., 1987.

[8] Wang, L. P. and Grandhi, R. V., "Efficient Safety Index Calculation for Structural Reliability Analysis", Computers and Structures, Vol. 52, No. 1, 1994, pp. 103-111.

[9] Wang, L. P., Grandhi, R. V., and Hopkins, D. A.,"Structural Reliability Optimization Using An Efficient Safety Index Calculation Procedure", International Journal for Numerical Methods in Engineering, Vol. 38, 1995, pp. 1721-1738.

[10] Wang, L. P. and Grandhi, R. V., "Improved Two-point Function Approximation for Design Optimization", AIAA Journal, Vol. 32, No. 9, 1995, pp. 1720-1727.

[11] VMA Engineering, DOT Users Manual, Version 4.00, 1993.

[12] Southwest Research Institute, "Probabilistic Structural Analysis Methods (PSAM) for Select Space Propulsion Systems Components", NESSUS Version 6.0 release notes, June 1992.

CHAPTER 4. FAILURE PROBABILITY CALCULATION

4.1 Monte Carlo Simulation

The Monte Carlo method was developed during World War II by Von Neumann and Ulam as a means of analyzing nuclear fission chain reactions. The application of the Monte Carlo method to probabilistic structural analysis problems is comparatively recent. This method became practical only with the advent of digital computers. It is a powerful mathematical tool for determining the approximate probability of a specific event that is the outcome of a series of stochastic processes. The Monte Carlo method consists of digital generation of random variables and functions, the statistical analysis of trial outputs, and variable reduction techniques. These are discussed briefly in this section.

In this section, the general principle of the Monte Carlo Simulation is introduced first (Section 4.1.1); and two key techniques needed in using the Monte Carlo method, such as the generation of uniformly distributed random numbers, and the generation of random variables which are given in Section 4.1.2 and Section 4.1.3, respectively. The direct Monte Carlo and the modification of Monte Carlo (importance sampling) are introduced in Sections 4.1.4 and 4.1.5, respectively.

4.1.1 General Principle of the Monte Carlo Simulation

As the name implies, Monte Carlo simulation techniques involve "sampling" at "random" to artificially simulate a large number of trials and observe the result. In the case of analysis for structural reliability, in the simplest approach, this means sampling each random variable x_i randomly gives a sample value $\hat{x}_i$. The limit state function $g(\hat{X}) = 0$ is then checked. If the limit state function is violated, the structure or structural element has "failed". The trial is repeated many times. In each trial, sample values are digitally generated and analyzed. If $\hat{N}$

trials are conducted, the probability of failure is given approximately by

$$P \approx \frac{m(g \leq 0)}{\hat{N}} \tag{4.1.1}$$

where m is the number of trials for which $g \leq 0$ out of the $\hat{N}$ experiments conducted.

In principle, the Monte Carlo simulation is only worth exploiting when the number of trials or simulations is less than the number of integration points required in numerical integration. This is achieved for higher dimensions by replacing the systematic selection of points by a "random" selection, under the assumption that the points selected will be in some way unbiased in their representation of the function being integrated.

Example 4.1

This example is taken from Ref. [1]. Some leaks were detected on the weld seam between the tubes and the tubesheet of a horizontal heat exchanger. To analyze the leakage events or failure that occurred in the heat exchanger, the following probable reasons are considered: (A) There was a preexisting crack at the time of manufacturing and it was not detected; (B) the crack grew to a critical size when unsteady operation resulted in fatigue; and (C) stress corrosion cracking (SCC) occurred. The probability of the occurrence of events A, B and C is P(A)=0.2, P(B)=0.15, P(C)=0.4, respectively. Random numbers generated for use in this example only are listed as 0.1, 0.09, 0.73, 0.25, 0.33, 0.76, 0.52, 0.01, 0.35, 0.86, 0.34, 0.67, 0.35, 0.48, 0.76, 0.80, 0.95, 0.90, 0.91, 0.17, 0.37, 0.54, 0.20, 0.48, 0.05, 0.64, 0.89, 0.47, etc. The probability of failure for the heat exchanger is estimated using the Monte Carlo method.

To solve the failure probability P_f, first assume that (1) the event does not occur if the value of random numbers generated is greater than the probability of occurrence for the event, and is denoted by 0; (2) the event occurs if the value of random numbers generated is smaller than the probability of occurrence for the event, and is denoted by 1. The simulation procedure is shown in Table 4.1. The trial is repeated 50 times ($\hat{N} = 50$) and 22 samples ($m = 22$) lead

to failure, so the failure probability can be computed approximately as

$$P_f \approx \frac{m}{\hat{N}} = \frac{22}{50} = 0.44$$

Since the probability of occurrence for each event is known, the exact result can be directly obtained from the following calculations:

(i) The probability that crack fatigue growth does not occur = 1-0.2 × 0.15 = 0.97

(ii) The probability that SCC does not occur = 1-0.4 = 0.6

(iii) The probability of failure = 1-0.97 × 0.6=0.418

Therefore, the probability of failure estimated using the Monte Carlo simulation is 5.26% in error compared to the exact result of 0.418

The simulation procedure illustrated in this example is the simplest Monte Carlo method for reliability problems; it may be the most widely used, but it is not the most efficient, especially for complex systems.

4.1.2 Generation of Uniformly Distributed Random Numbers

To use the Monte Carlo simulation to solve a practical problem, it is necessary to generate random numbers for different distributions. The random number of the [0,1] interval uniformly distributed is the simplest and most important random number. Based on this random number, the random number with arbitrary probability distributions can be obtained by means of various sampling techniques. Therefore, the [0,1] interval uniformly distributed random number is the basis for generating various distributed random numbers.

Assuming that x is the random variable over the interval [0,1], its density function is given as

$$f(x) = \begin{cases} 1, & 0 \le x \le 1 \\ 0, & otherwise \end{cases} \tag{4.1.2}$$

Table 4.1. The Monte Carlo Simulation Procedure for Example 4.1

Trial number	Cause for leakage	Random number			Event occurrence				
		A	B	C	A	B	C	Leakage	Failure
1	Crack fatigue growth	0.1	0.09		1	1		1	1
	SCC			0.73			0	0	
2	Crack fatigue growth	0.25	0.33		0	0		0	0
	SCC			0.76			0	0	
3	Crack fatigue growth	0.52	0.01		0	1		0	1
	SCC			0.35			1	1	
4	Crack fatigue growth	0.86	0.34		0	0		0	0
	SCC			0.67			0	0	
.....									

and its distribution function is given as

$$F(x) = \begin{cases} 0, & x < 0, \\ x, & 0 \le x \le 1 \\ 1, & x > 1 \end{cases} \tag{4.1.3}$$

The sampling sequence $\hat{r}_1$, $\hat{r}_2$, ..., $\hat{r}_{\hat{n}}$, ... over the uniformly distributed random variable x over the interval [0,1] is called the random number of the [0,1] interval uniformly distributed.

There are many methods available for generating the random numbers. The most common practical approach is to employ a suitable "pseudo" random number generator (PRNG), available on virtually all computers of numbers. They are "pseudo" since they use a formula to generate a sequence of numbers. This sequence is reproducible and repeats normally after a long cycle interval. The following recurrence formula is usually used to generate the "pseudo" random number.

$$x_{\hat{n}+1} = \gamma x_{\hat{n}} (mod\ M) \tag{4.1.4a}$$

$$\hat{r}_{\hat{n}+1} = \frac{x_{\hat{n}+1}}{M} \tag{4.1.4b}$$

where γ, M and x_0 are the preselected positive integers, which are determined by semi-theoretical and semi-experiential approaches.

Eq. (4.1.4) means that $x_{\hat{n}+1}$ is the remainder of the product $\gamma x_{\hat{n}}$ dividing by M, the value from $x_{\hat{n}+1}$ dividing by M is the $(\hat{n}+1)$ uniformly distributed random number, $\hat{r}_{\hat{n}+1}$. Repeating the process, a sequence of random numbers can be obtained.

For most practical purposes, a sequence of numbers generated by a suitable modern PRNG is indistinguishable from a sequence production of a reproducible sequence, which can be an advantage in certain problems. However, this reproducibility can be destroyed simply by (randomly) changing the "seed number" required for most PRNGs. A simple solution is to use the local time as a seed value.

4.1.3 Generation of Random Variables

One of the key features in the Monte Carlo method is the generation of a series of values of one or more random variables with specified probability distributions. The most commonly-used generation method is the "inverse transform" method. Let $F_X(x_i)$ be the cumulative function of random variable x_i. By definition, the numerical value of $F_X(x_i)$ is a value in the interval of [0,1]. Assuming that ξ_i is the generated uniformly distributed random number $(0 \leq \xi_i \leq 1)$, the inverse transform method is used to equate the ξ_i to $F_X(x_i)$ as follows:

$$F_X(x_i) = \xi_i \quad or \quad x_i = F_X^{-1}(\xi_i) \tag{4.1.5}$$

This method can be applied to variables for which a cumulative distribution function has been obtained from direct observation, or an analytic expression for the inverse cumulative function, $F^{-1}(.)$, exists.

Example 4.2

Generate random variables with type-I extreme value distribution.

The probability density function of the type-I extreme value distribution is

$$f_x(x) = \frac{1}{t_0} exp\{-(\frac{x-\delta}{t_0}) - exp[-(\frac{x-\delta}{t_0})]\}$$

The cumulative distribution function is

$$F_x(x) = exp[-exp(-\frac{x-\delta}{t_0})]$$

where t_0 is a scale parameter and δ is a location parameter.

Let ξ be the random numbers from uniform distribution over the interval [0,1], and η be the random variable with the specified distribution. Based on Eq. (4.1.5),

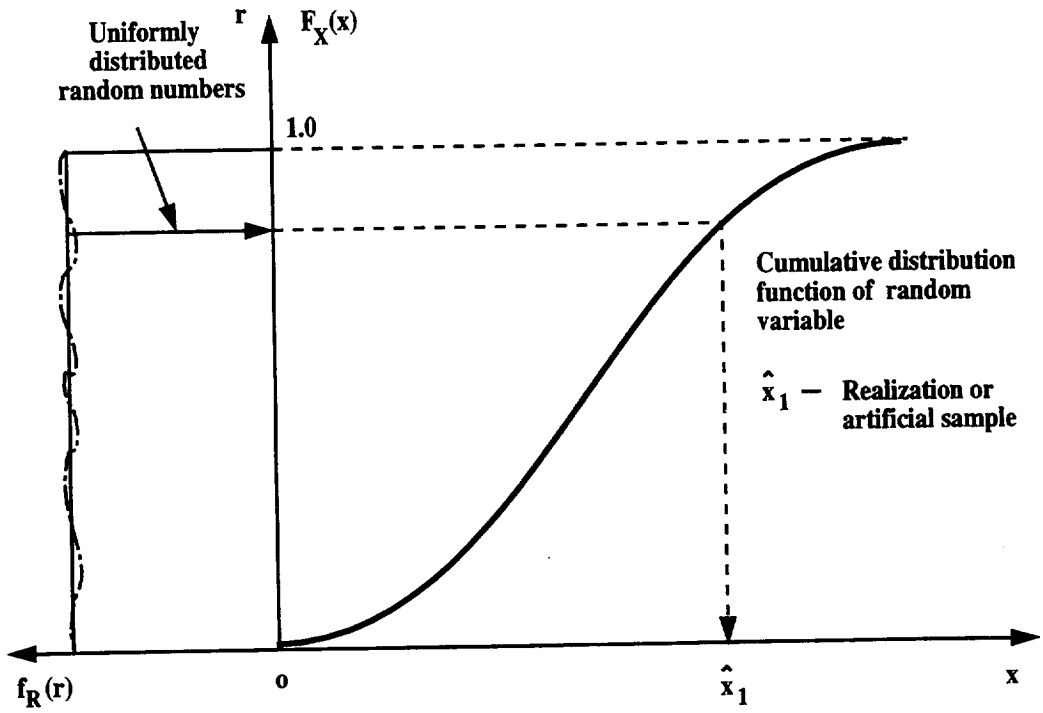


Fig. 4.1 Inverse transform method for generation of random variates

$$F_x(\eta) = exp[-exp(-\frac{\eta - \delta}{t_0})] = \xi$$

hence

$$\eta = \delta - t_0 ln(-ln\xi)$$

The inverse transform technique is shown in Fig. 4.1.

4.1.4 Direct Sampling (Crude Monte Carlo)

Eq. (4.1.1) gives the simplest Monte Carlo approach for reliability problems; it may be the most widely used, but it is not the most efficient, especially in complex systems. A commonly-used technique called crude Monte Carlo is introduced below.

As defined in Eq. (3.1.3) in Section 3.1, the failure region is the event that the limit state $g(X) \leq 0$, and the failure probability

$$P_f = P[g(X) \leq 0] = \int \ldots \int_{g(X)\leq 0} f_X(X)dX \tag{4.1.6a}$$

is rewritten as

$$P_f = \int \ldots \int I[g(X) \leq 0] f_X(X)dX \tag{4.1.6b}$$

where $I[.]$ is an "indicator function" which equals 1 if [.] is "true" and 0 if [.] is "false".

In comparison with (B.2), it can be seen that the expectation of the indicator random variable for the failure event is just the probability that failure occurs. Hence

$$P_f = P[g(X) \leq 0] = E[I(X)] = \mu_I = \mu_{P_f} \tag{4.1.7}$$

and its variance is

$$\begin{aligned} Var[I(X)] &= E[I(X)^2] - \{E[I(X)]\}^2 \\ &= E[I(X)] - \{E[I(X)]\}^2 \\ &= E[I(X)]\{1 - E[I(X)]\} \\ &= P_f(1 - P_f) \end{aligned} \tag{4.1.8}$$

In order to evaluate P_f by the Monte Carlo method, a sample value for basic variable x_i with a cumulative distribution $F_x(x_i)$ must be drawn. The "inverse transform" method given in Section 4.1.3 can be used to obtain the random variate, in which a uniformly distributed random number ξ_i $(0 \leq \xi_i \leq 1)$ is generated and is equated to $F_x(x_i)$, *i.e.*, $x_i = F_x^{-1}(\xi_i)$.

Hence, independent random numbers $\xi_1, \xi_2, \ldots, \xi_{\hat{n}}$ are drawn from the density $f_x(x_i)$ and the estimate of P_f is obtained:

$$\hat{P}_f = \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} I(\xi_i) \tag{4.1.9}$$

where $\hat{p}_f$ represents the crude Monte Carlo estimator of μ_{P_f} and $\hat{n}$ is the number of the independent random numbers.

The variance of the sample mean is computed as

$$Var(\hat{P}_f) = \frac{\sigma_I^2}{\hat{n}} = \sigma_{P_f}^2 \tag{4.1.10}$$

So the sample variance is given as

$$S_I^2 = \frac{1}{\hat{n}-1}\{\sum_{i=1}^{\hat{n}} I^2(\xi_i) - \hat{n}[\frac{1}{\hat{n}}\sum_{i=1}^{\hat{n}} I(\xi_i)]^2\} \tag{4.1.11}$$

Example 4.3

This example is taken from Ref. [1]. The limit state function is given as

$$g(X) = R - S = 0$$

where R and S represent strength and stress, respectively. Both R and S are normally distributed with the following means and standard deviations:

$$\mu_R = 135MPa, \quad \sigma_R = 15.0MPa, \qquad \mu_S = 100MPa, \quad \sigma_S = 12.0MPa$$

Estimate the failure probability using the crude Monde Carlo method.

For convenience, only 14 variates $\hat{R}$ and another 14 variates $\hat{S}$ are given below, but generally more samples are needed. The procedure of the Monte Carlo simulation is shown in Table 4.2, where

$$\hat{R} = \mu_R + \sigma_R \Phi^{-1}(\xi_i)$$

$$\hat{S} = \mu_S + \sigma_S \Phi^{-1}(\xi_i)$$

ξ are the random numbers given in Example 4.1. In Table 4.2, only one of the sample pairs ($\hat{R}$, $\hat{S}$) led to a failure (i.e. $\hat{R}_i \leq \hat{S}_i$).

The failure probability is

$$\hat{P}_f == \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} I(\xi_i) = \frac{1}{14} = 0.07143$$

Since both R and S are normally distributed, the exact result can be calculated as

$$P_f = \Phi(-\frac{132 - 100}{\sqrt{15^2 + 12^2}}) = \Phi(-1.666) = 0.04746$$

Obviously, more sampling is required.

4.1.5 Variance Reduction - Importance Sampling

Variance reduction techniques have a dual purpose: to reduce the length of a sample run and to increase accuracy using the same number of runs.

In structural reliability analysis, where the probability of failure is generally relatively small, the direct (crude) Monte Carlo simulation procedure becomes inefficient. For example, in many pressure vessel technology problems, the probability of failure could be as small as 10^{-5} or 10^{-10}; this implies that nearly a million simulation repetitions are required to predict this behavior. If the limit state function $g(X)$ represents the mathematical model for structural simulation problems, the tail of the distribution of $g(X)$ is the most important factor. The simulated iteration must concentrate the sample points in this part in order to predict the risk reliably and to increase the efficiency of the simulation by expediting execution and minimizing computer storage requirements. Slow convergence is a severe penalty for the direct Monte Carlo method and has led to several variance reduction techniques. The importance sampling method, systematic sampling method, stratified sampling method, split sampling method, Latin hypercube sampling method, conditional expectation method, and antithetic variates method are some of the popular variance reduction techniques. Here, the importance sampling method is briefly introduced as an illustration to interpret the concept of variance reduction techniques.

The importance sampling method is a modification of Monte Carlo simulation in which the simulation is biased for greater efficiency. In importance sampling, the sampling is done only

Table 4.2. The Monte Carlo Simulation Procedure for Example 4.3

Strength			Stress			$I(\xi_i)$
ξ_i	$\Phi^{-1}(\xi_i)$	$\hat{R}$	ξ_i	$\Phi^{-1}(\xi_i)$	$\hat{S}$	
0.1	-1.28	112.8	0.76	0.71	108.52	0
0.09	-1.34	111.9	0.80	0.84	111.08	0
0.73	0.61	159.2	0.95	1.64	119.68	0
0.25	-0.67	121.95	0.90	1.28	115.36	0
0.33	-0.44	125.4	0.91	1.34	116.08	0
0.76	0.71	142.65	0.17	-0.95	88.6	0
0.52	0.05	132.75	0.37	-0.33	96.04	0
0.01	-2.33	97.05	0.54	0.1	101.2	1
0.35	-0.39	126.15	0.20	-0.84	89.92	0
0.86	1.08	148.2	0.48	-0.05	99.4	0
0.34	-0.42	125.7	0.05	-1.64	80.32	0
0.67	0.44	138.6	0.64	0.36	104.32	0
0.35	-0.39	126.15	0.89	1.22	114.64	0
0.48	-0.05	131.25	0.47	-0.08	99.04	0

in the tail of the distribution instead of spreading samples out evenly in order to ensure that simulated failure occurs.

The failure probability of Eq. (4.1.6b) can be rewritten as

$$P_f = P[g(X) \leq 0] = \int \ldots \int \frac{I[X] f_X(X)}{f_X^*(X)} f_X^*(X) dX \tag{4.1.12}$$

where $f_X^*(X)$ is the importance sampling probability density function. In comparison to Eq. (A.3.2), the expectation of the indicator function in Eq. (4.1.12) can be written in the form

$$P_f = E[\frac{I(X) f_X(X)}{f_X^*(X)}] = \mu_{P_f} \tag{4.1.13}$$

Let $x_1^*, x_2^*, \ldots, x_{\hat{n}}^*$ denote random observations from the importance sampling function, $f_X^*(.)$. An unbiased estimate of P_f is given by

$$\hat{P}_f^* = \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} \frac{I(x_i^*) f_x(x_i^*)}{f_x^*(x_i^*)} \tag{4.1.14}$$

The choice of $f_x^*(.)$ is quite important. If the density $f_x^*(.)$ has been chosen so that there is an abundance of observations for which $I(x_i^*) = 1$ and if the ratio $f_x(x_i^*)/f_x^*(x_i^*)$ does not change much with different values of x_i^*, then $Var[\hat{P}_f^*]$ will be much less than $Var[\hat{P}_f]$ (Eq. 4.1.10). Consequently, $\hat{P}_f^*$ requires many fewer observations than $\hat{P}_f$ (Eq. 4.1.9) to achieve the same degree of precision.

The variance of $\hat{P}_f^*$ is given by

$$\begin{aligned} Var[\hat{P}_f^*] &= \frac{1}{\hat{n}} \{ \int \ldots \int [\frac{I(x) f_x(x)}{f_x^*(x)}]^2 f_x^*(x) dx - \mu_{P_f}^2 \} \\ &= \frac{1}{\hat{n}} \{ \int \ldots \int \frac{[I(x)]^2 [f_x(x)]^2}{f_x^*(x)} dx - \mu_{P_f}^2 \} \end{aligned} \tag{4.1.15}$$

The ideal choice of $f_x^*(x)$ is obtained using calculus.

$$\frac{\partial}{\partial [f_x^*(x)]} \{ Var[\hat{P}_f^*] + \lambda [\int \ldots \int f_x^*(x) dx - 1] \} = 0 \tag{4.1.16}$$

where λ is a Lagrangian multiplier. This can be solved by using the calculus of variations. We obtain

$$f_x^*(x) = \frac{|I(x)f_x(x)|}{\int \ldots \int |I(x)f_x(x)|dx} \tag{4.1.17}$$

Substituting into Eq. (4.1.15), it is easily found that

$$Var[\hat{P}_f^*] = \frac{1}{\hat{n}}\{[\int \ldots \int |I(x)f_x(x)|dx]^2 - \mu_{P_f}^2\} \tag{4.1.18}$$

If $|I(x)f_x(x)|$ is positive everywhere, the multiple integral is identical with μ_{P_f} and $Var[\hat{P}_f^*] = 0$. In this case the optimal function $f_x^*(.)$ is

$$f_x^*(x) = \frac{I(x)f_x(x)}{\mu_{P_f}} \tag{4.1.19}$$

It can be seen that a good choice of $f_x^*(.)$ can produce zero variance. Since μ_{P_f} is unknown, this is impossible. However, it demonstrates that if more effort is put into obtaining a close initial estimate of P_f, then the $Var(\hat{P}_f^*)$ will be much less than the variance of $\hat{P}_f$ in Eq.(4.1.9). Conversely, the variance can actually be increased using a very poor choice of $f_x^*(.)$. Thus, the application of importance sampling is sometimes referred to as an art which must be used with caution.

4.2 Response Surface Approximation

The multi-dimensional integration of Eq. (4.1.6a) may be carried out analytically for a very limited number of cases. For most practical cases, the integration is impossible to conduct analytically. Numerical methods, such as the Monte Carlo simulation, can generally be performed to evaluate the integration, but it often turns out to be too computer time consuming. Thus, relatively simple and accurate approximate techniques to evaluate the reliability are needed.

Chapter 3 has introduced some approximate methods for the safety index calculations, which is the basis of the failure probability calculations given in this section. Even though the safety index can be used for the measure of the structural safety, we are still more interested in the reliability of structures, *i.e.*, the failure probability.

Since the n-dimensional probability density function can be given as a multiplication of n probability density functions for n random variables with standard normal distributions in the standard normal distribution space, most approximate methods for failure probability calculations are generated in the standard normal space so that the n-dimensional integration can be easily computed. Once the MPP U^* is located using the safety index algorithms given in Chapter II, the approximate failure probability can be calculated by (i) approximating the limit state surface $g(U)$ using an approximate surface at the MPP; and (ii) evaluating the failure probability defined by the approximating surface by exact or approximate means.

Different approximate response surfaces result in different methods of the failure probability calculations. If the response surface is approached by a first-order approximation at the MPP, the method is called first-order reliability method (FORM); if the response surface is approached by a second-order approximation at the MPP, the method is called second-order reliability method (SORM). Furthermore, if the response surface is approached by a higher-order approximation at MPP, the method is called higher-order reliability method (HORM). Hence, the response surface approximations play an important role in the failure probability calculations.

In this section, several different approximations which are used in FORM, SORM and HORM are introduced. Since the approximations are obtained in a new rotated space, the orthogonal transformation from U-space to the new space is introduced first in Section 4.2.1. The first-order and second-order Taylor approximations are given in Sections 4.2.2 and 4.2.3. The adaptive nonlinear approximation is introduced in Section 4.2.4.

4.2.1 Orthogonal Transformations

In most failure probability calculations, the integration of Eq. (4.1.6a) is performed in the rotated new standard normal Y-space instead of U-space. In order to conduct the rotation from the standard normal U-space to the Y-space, an orthogonal matrix H needs to be generated, in which the nth row of H is the unit normal of the limit state function at the MPP, i.e., $-\nabla G(U^*)/|\nabla G(U^*)|$. To generate H, first, an initial matrix is selected as follows

$$\begin{pmatrix} \frac{-\partial G(U^*)/\partial U_1}{|\nabla G(U^*)|} & \frac{-\partial G(U^*)/\partial U_2}{|\nabla G(U^*)|} & \cdots & \frac{-\partial G(U^*)/\partial U_n}{|\nabla G(U^*)|} \\ 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \tag{4.2.1}$$

where the last $n-1$ rows consist of zeros and unity on the diagonal. The Gram-Schmidt algorithm [2] is used to orthogonalize the above matrix to obtain an orthogonal matrix. First, let f_1, f_2, ..., f_n denote the first, second, ..., nth row vector of the above matrix, respectively; *i.e.*,

$$f_1 = \{\frac{-\partial G(U^*)/\partial U_1}{|\nabla G(U^*)|}, \frac{-\partial G(U^*)/\partial U_2}{|\nabla G(U^*)|}, ..., \frac{-\partial G(U^*)/\partial U_n}{|\nabla G(U^*)|}\}^T$$

$$f_2 = \{0, 1, 0, ..., 0\}^T$$

.....

$$f_n = \{0, 0, 0, ..., 1\}^T$$

Set

$$D_1 = (f_1, f_1)^{\frac{1}{2}}$$

$$e_{11} = \frac{1}{D_1}$$

$$\gamma_1 = e_{11} f_1$$

$$D_2 = [(f_2, f_2) - |(f_2, \gamma_1)|^2]^{\frac{1}{2}}$$

$$e_{12} = -\frac{(f_2, \gamma_1)}{D_2}$$

$$e_{22} = \frac{1}{D_2}$$

$$\gamma_2 = e_{12}\gamma_1 + e_{22}\gamma_2$$

and, in general,

$$D_k = [(f_k, f_k) - |(f_k, \gamma_1)|^2 - |(f_k, \gamma_2)|^2 -, ..., -|(f_k, \gamma_{k-1})|^2]^{\frac{1}{2}}$$

$$e_{1k} = -\frac{(f_k, \gamma_1)}{D_k}$$

$$e_{2k} = -\frac{(f_k, \gamma_2)}{D_k}$$

.....

$$e_{k-1,k} = -\frac{(f_k, \gamma_{k-1})}{D_k}$$

$$e_{kk} = \frac{1}{D_k}$$

$$\gamma_k = e_{1k}\gamma_1 + e_{2k}\gamma_2 +, ..., e_{k-1,k}\gamma_{k-1} + e_{kk}\gamma_k$$

where (f, f) and (f, γ) represent the scalar product (dot product) of two vectors. It can be verified that the generated vectors γ_1, γ_2, ..., γ_n are orthogonalized. The generated orthogonal matrix H_0 is

$$H_0 = \begin{pmatrix} \gamma_1{}^T \\ \gamma_2{}^T \\ \vdots \\ \gamma_n{}^T \end{pmatrix} \tag{4.2.2}$$

In fact, in the orthogonal matrix of Eq. (4.2.2), the first row is $-\nabla G(U^*)/|\nabla G(U^*)|$ due to $D_1 = 1$. To satisfy that the nth row of H is $-\nabla G(U^*)/|\nabla G(U^*)|$, the first row of the orthogonal matrix is moved to the last row. This rearranged matrix is also an orthogonalized

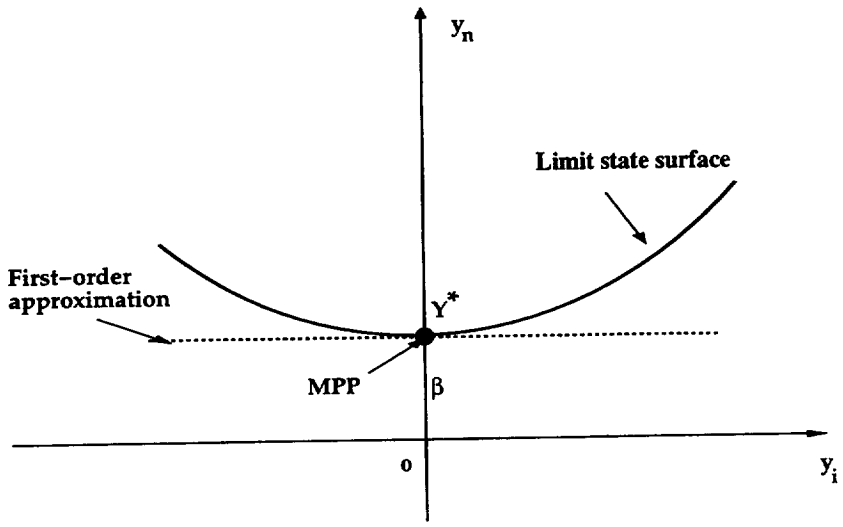


Fig. 4.2 First–order approximation of the response surface in Y–space

matrix and satisfies that the nth row of H equals $-\nabla G(U^*)/|\nabla G(U^*)|$, so it is defined as H matrix and is given as

$$H = \begin{pmatrix} {\gamma_2}^T \\ {\gamma_3}^T \\ \vdots \\ {\gamma_n}^T \\ {\gamma_1}^T \end{pmatrix} \tag{4.2.3}$$

4.2.2 First-order Approximation of Response Surfaces

Assuming the most probable failure point (MPP) in U-space as $U^* = \{u_1^*, u_2^*, ..., u_n^*\}^T$, the linear approximation of the response surface $g(U) = 0$ is given by the first-order Taylor series

expansion at the MPP,

$$\tilde{g}(U) \approx g(U^*) + \nabla g(U^*)(U - U^*) = 0 \tag{4.2.4}$$

In this equation, $g(U^*)$ equals 0 because U^* is on the response surface. Divided by $|\nabla g(U^*)|$, Eq. (4.2.4) is rewritten as

$$\tilde{g}(U) \approx \frac{\nabla g(U^*)}{|\nabla g(U^*)|}(U - U^*) \tag{4.2.5}$$

From Eq. (3.3.25), we have

$$\frac{\nabla g(U^*)U^*}{|\nabla g(U^*)|} = -\beta \tag{4.2.6}$$

Substituting this equation into Eq. (4.2.5), we obtain

$$\tilde{g}(U) \approx \frac{\nabla g(U^*)}{|\nabla g(U^*)|}U + \beta = 0 \tag{4.2.7}$$

By a rotation of U into a new set of mutually independent standard normal random variables Y using the orthogonal matrix H given in Eq. (4.2.3),

$$Y = HU \tag{4.2.8}$$

and the approximate response surface given in Eq. (4.2.7) becomes

$$\tilde{g}(U) \approx -y_n + \beta = 0 \tag{4.2.9a}$$

or

$$y_n = \beta \tag{4.2.9b}$$

Eq. (4.2.9b) is the first-order approximation of the response surface in the rotated standard normal space (denoted as Y-space), as shown in Figure 4.2. If the limit state functions of the practical problems are linear or close to linear, this approximation closely or exactly represents

the response surface. Otherwise, the truncation errors from the first-order Taylor approximation might be large and more accurate approximations would be needed.

4.2.3 Second-order Approximation of Response Surfaces

The second-order approximation of the response surface $g(U) = 0$ is given by the second-order Taylor series expansion at the MPP,

$$\tilde{g}(U) \approx g(U^*) + \nabla g(U^*)(U - U^*) + \frac{1}{2}(U - U^*)^T \nabla^2 g(U^*)(U - U^*) \tag{4.2.10}$$

where $\nabla^2 g(U^*)$ represents the symmetric matrix of the second derivatives of the failure function:

$$\nabla^2 g(U^*)_{ij} = \frac{\partial^2 g(U^*)}{\partial u_i \partial u_j} \tag{4.2.11}$$

Dividing by $|\nabla g(U^*)|$ and considering $g(U^*)$=0, we obtain

$$\tilde{g}(U) \approx \alpha^T (U - U^*) + \frac{1}{2}(U - U^*)^T B (U - U^*) \tag{4.2.12a}$$

where

$$\alpha = \frac{\nabla g(U^*)}{|\nabla g(U^*)|} \tag{4.2.12b}$$

and

$$B = \frac{\nabla^2 g(U^*)}{|\nabla g(U^*)|} \tag{4.2.12c}$$

Physically, the following transformations are the coordinate rotations to make the y_n axis coincide with the β vector, as shown in Figure 4.3. Substituting Eq. (4.2.8) into Eq. (4.2.12) and replacing the first term by Eq.(4.2.9a), the U-space approximate response surface is rotated as

$$\tilde{g}(Y) \approx -y_n + \beta + \frac{1}{2}(H^{-1}Y - H^{-1}Y^*)^T B (H^{-1}Y - H^{-1}Y^*) \tag{4.2.13}$$

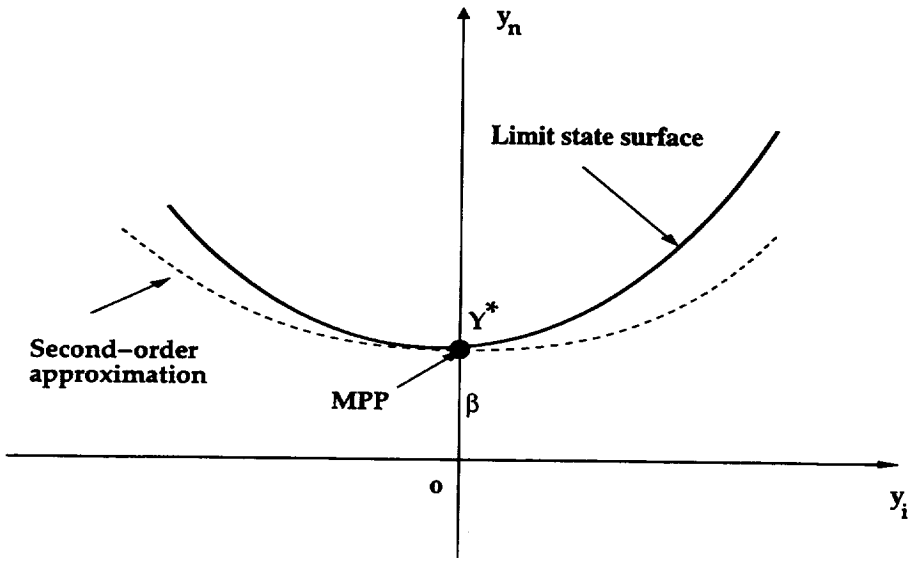


Fig. 4.3 Second–order approximation of the response surface in Y–space

where Y^* is the Y-space MPP ($Y^* = \{0, 0, ..., \beta\}^T$) corresponding to the U-space MPP U^*. In Y-space, y_n axis is in coincidence with the β vector.

Since the H matrix is an orthogonal matrix,

$$H^{-1} = H^T \tag{4.2.14}$$

Substituting this equation into Eq. (4.2.13), we have

$$\tilde{g}(Y) \approx -y_n + \beta + \frac{1}{2}(Y - Y^*)^T HBH^T(Y - Y^*) \tag{4.2.15}$$

where

$$(Y - Y^*)^T = (y_1, y_2, ..., y_n - \beta)^T \tag{4.2.16}$$

By a series of orthogonal transformations, H_1, H_2, ...H_m, for the first $n-1$ variables, $\bar{Y} = \{y_1, y_2, ..., y_{n-1}\}^T$, *i.e.*,

$$\bar{Y}' = H_1 H_2, ...H_m \bar{Y} \tag{4.2.17}$$

Quantities associated with $n-1$ variables are denoted with a bar. Finally, the first $(n-1) \times (n-1)$ order matrix of HBH^T will become a diagonal matrix

$$\bar{H}\bar{B}\bar{H}^T = \begin{pmatrix} \kappa_1 & 0 & \dots & 0 \\ 0 & \kappa_2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \kappa_{n-1} \end{pmatrix} \tag{4.2.18}$$

and Eq. (4.2.15) becomes

$$y_n = \beta + \frac{1}{2}\sum_{i=1}^{n-1} \kappa_i y_i'^2 \tag{4.2.19}$$

In fact, the above procedure of finding the diagonal matrix can be treated as an eigenvalue problem. So κ_i are the eigenvalues of matrix $\bar{H}\bar{B}\bar{H}^T$ whose elements are given by

$$\kappa_{ij} = (\bar{H}\bar{B}\bar{H}^T)_{ij}, \qquad i,j = 1,2,...,n-1 \tag{4.2.20}$$

where κ_{ij} represents the curvature of the response surface at the MPP.

Eq. (4.2.19) is the second-order approximation of the response surface in the rotated standard normal space. The major computational cost is in computing the second derivatives B of the limit state function at MPP. The exact second-order derivatives of $g(U)$ require additional $n(n+1)/2$ function calls for a finite difference scheme. For problems having a large number of random variables, this calculation is extremely computer intensive. From this procedure, it is clear that one has to increase the computational efficiency in calculating the curvature matrix and second-order function derivatives. Then it enables an accelerated and cost-effective

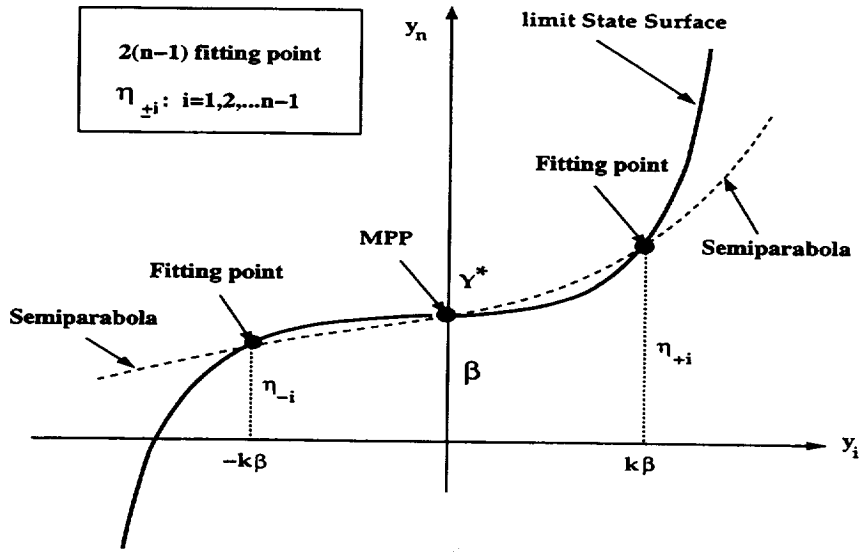


Fig. 4.4 Fitting of paraboloid in Y–space

procedure to perform the second-order probability analysis, particularly when finite-element-based structural analysis tools are used. In the next section, two second-order approximations, which were presented by Wang and Grandhi [3] and by Der Kiureghian, *et al* [4], with no computational cost of the second-order derivatives calculations are introduced.

4.2.4 Second-order Approximation of Response Surfaces with Approximate Curvature

In Ref. [3], the second-order derivatives of the limit state function are approximately calculated by using an approximate performance function. This approximate performance function was constructed during the safety index search process, which is given in Eq. (3.3.74). The approximate second-derivatives are given as

$$\begin{aligned}\nabla^2\tilde{g}(U^*)_{ij} &= \frac{\partial^2\tilde{g}(U^*)}{\partial u_i \partial u_j} \\ &= (r-1)\sum_i^n (u_{i,k}+\frac{\bar{x}_i}{\sigma_i}s)^{(1-r)}\frac{\partial g(Y_k)}{\partial u_i}(u_i+\frac{\bar{x}_i}{\sigma_i}s)^{r-2}\nabla^2\tilde{g}(U^*)_{ij} \\ &= 0, \qquad i \neq j \end{aligned} \tag{4.2.21}$$

By considering Eq. (4.2.12c) and the orthogonal transformation of Eq. (4.2.8), the curvature κ_i can be approximately determined from Eq. (4.2.20). Since the nonlinear function given in Eq. (3.3.52) is fairly accurate around the MPP when convergence is realized, the calculation of second-order derivatives using this nonlinear approximation would give improved accuracy in failure probability compared to the first-order methods. Also, this procedure avoids the exact second-order derivatives computations of the limit state function at the MPP.

In Ref. [4], the approximating paraboloid is defined by fitting a set of discrete points selected on the limit state surface at prescribed distances from the MPP. These fitting points, $2(n-1)$ in number, are selected along the coordinate axes in the rotated space in the manner described in Figure 4.4. Along each axis y_i, $i = 1, 2, ..., n-1$, two points are selected with the coordinates $(-k\beta, \eta_{-i})$ and $(k\beta, \eta_{+i})$, where the subscripts $-i$ and $+i$ refer to the negative and positive directions on the y_i axis, respectively, and k is a preselected coefficient. The ordinates $\eta_{\pm i}$ are obtained as solutions of y_n in $g(Y) = 0$ with $Y = (0, ..., \pm k\beta, 0, ..., y_n)$. A simple algorithm for finding these solutions is described in the Appendix C.

This method is expected to provide computational savings when the number of variables is very large. In fact, if finding the ordinates $\eta_{\pm i}$ needs m iterations, this procedure requires $2 \times n \times m$ exact g-function calculations. If n is large and m is about 10 to 20, then this procedure may not be efficient even though it might take less computational time than the original second-order approximation. An advantage of this method is that it can be used for problems with an inflection point at Y^* since two semiparabolas are used.

In all the mentioned second-order approximations, if the limit state functions of the practical

problems are quadratic, linear, or close to quadratic or linear, the approximation would fit the response surface well. However, in some cases, the limit state functions may be highly nonlinear or have an inflection point at the MPP (see Figure 4.5 in the next section). For these cases, the second-order approximation might result in larger errors. Hence, a more accurate and adaptive approximation is given in the next section, which was presented by Wang and Grandhi in [5].

4.2.5 Higher-order approximation of response surfaces

As discussed in the previous section, in some cases, the second-order approximation can not yield a good representation for the response surface because the curvatures at the MPP do not provide a realistic picture of the surface in the neighborhood of the MPP. For example, when the MPP is an inflection point, as shown in Figure 4.5a or when the limit state surface is highly nonlinear, as shown in Figure 4.5b, the curvatures are zero and the second-order approximate surface reduces to the tangent plane, thus, providing no improvement over the first-order approximation.

In order to obtain the higher-order approximation of the response surface, first, we need to rotate the standard normal U-space to the rotated standard normal Y-space. The procedure of the rotation transformation is given in Eq. (4.2.8). In Y-space, the random variables Y are also independent, standard and normally distributed. The y_n axis is in coincidence with the β vector. In order to obtain a more accurate approximation in Y-space, an adaptive nonlinear function is constructed based on two points selected on the limit state surface with the coordinates $(-k\beta,\ -k\beta,\ ....,\ \eta_a)$, and $(k\beta, k\beta, ..., \eta_b)$, where η_a and η_b are the y_n values corresponding to the negative and positive directions of the y_i $(i = 1, 2, ..., n-1)$, respectively. k is a preselected coefficient which can be selected from 0.1 to 1.0. In this work, k is selected as 0.1. The ordinates η_a and η_b are obtained as the solutions of y_n in $g(Y) = 0$ with $Y_a = (-k\beta,\ -k\beta, ...\ -k\beta,\ \eta_a)$, and $Y_b = (k\beta,\ k\beta, ...\ k\beta,\ \eta_b)$. A simple algorithm for finding η_a and η_b is given in Appendix C. Unlike in Ref. [4], the iterations for finding the coordinates η_a and η_b are

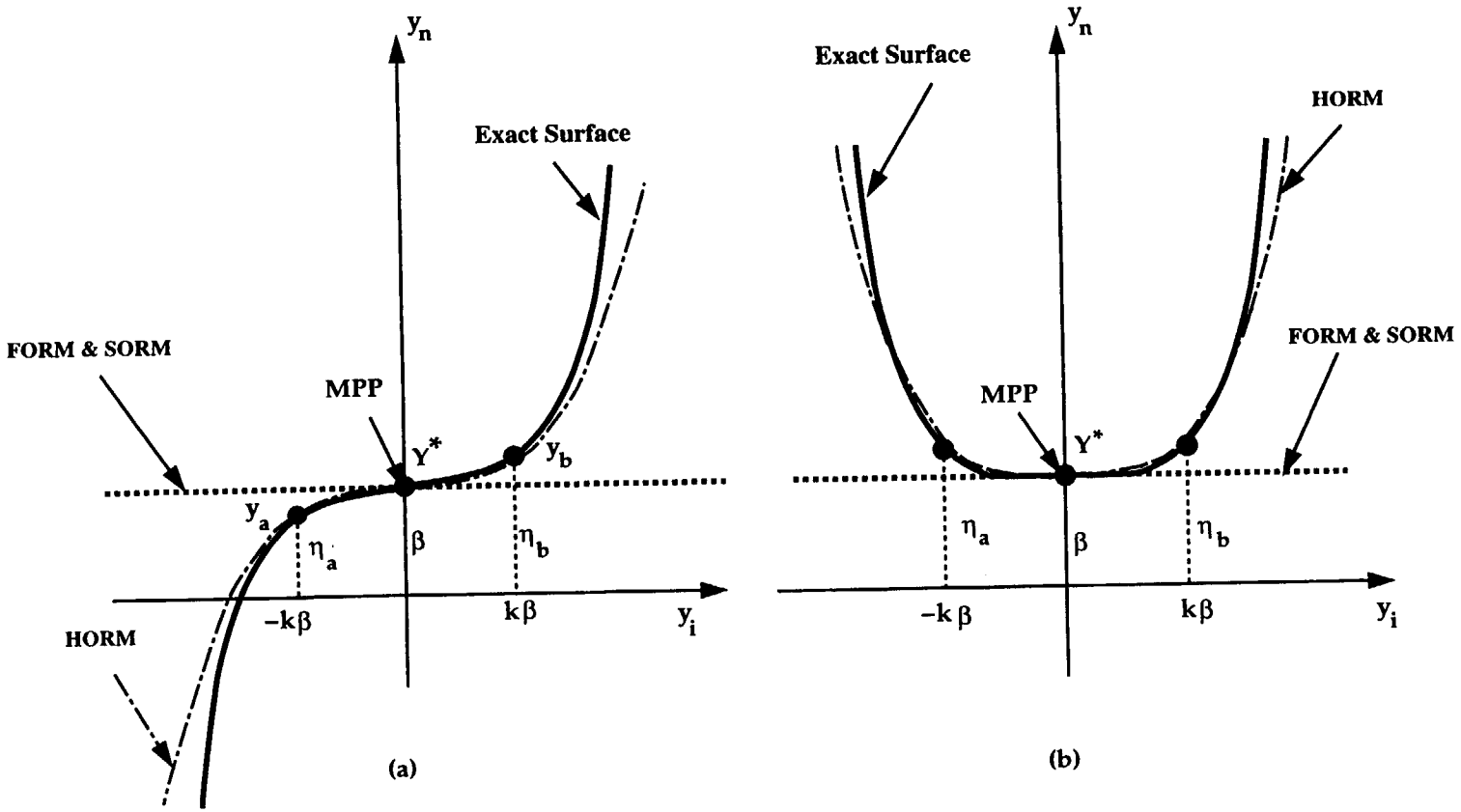


Fig. 4.5 Higher–order and other approximations for the response surface

performed on the approximate performance function instead of the exact analyses, which was constructed for computing the safety index β. This nonlinear approximation is fairly accurate in the neighborhood of the MPP when the convergence for β is realized. These approximations provide tremendous computational savings when the finite element methods are used for the analyses.

Based on these two points selected on the limit state surface, a two-point adaptive nonlinear approximation is established. The approximation is similar to the one used in finding the MPP in Ref. [6] but the nonlinear index is an integer instead of a real number. The errors from the integer index are eliminated by adding the weight coefficients to each term of the approximation. The approximation is required to pass through two base points and the MPP. The details of the approximation based on Y_a and Y_b are given as follows. The intervening variables are defined as

$$s_i = y_i^m \qquad i = 1, 2, ..., n-1 \tag{4.2.22}$$

where m is the nonlinearity index, which can be a positive or negative. Since the random variables y_i $(i = 1, 2, ..., n-1)$ vary from $-\infty$ to $+\infty$ in Y-space, the nonlinearity index m is defined as an integer to avoid numerical problems in Eq. (4.2.22) when y_i is negative with a real index. The approximate function $\tilde{y}_n$ is obtained by expanding the function in terms of the intervening variables given in Eq. (4.2.22) at $\bar{Y}_a = (-k\beta, -k\beta, ..., -k\beta)$. The weight coefficients are added to each term of the approximation to improve the accuracy since the nonlinearity index is an integer. The approximation is written as

$$\tilde{y}_n(\bar{Y}) = y_n(\bar{Y}_a) + \frac{1}{m}\sum_{i=1}^{n-1} t_i y_{i,a}^{1-m} \frac{\partial y_n(\bar{Y}_a)}{\partial y_i}(y_i^m - y_{i,a}^m) \tag{4.2.23a}$$

where $\tilde{y}_n(\bar{Y})$ and $y_n(\bar{Y})$ represent the approximate and exact functions, respectively, and $\frac{\partial y_n(\bar{Y}_a)}{\partial y_i}$ can be computed by the differential method of implicit functions as

$$\frac{\partial y_n(\bar{Y}_a)}{\partial y_i} = -\frac{\partial g(Y_a)}{\partial y_i} \Big/ \frac{\partial g(Y_a)}{\partial y_n} \tag{4.2.23b}$$

where $\bar{Y} = \{y_1, y_2, ..., y_{n-1}\}^T$ and $Y = \{y_1, y_2, ..., y_n\}^T$. The function value at the MPP, i.e. $Y^*(0, 0, ..., \beta)$, is

$$\beta = y_n(\bar{Y}_a) + \frac{1}{m}\sum_{i=1}^{n-1} t_i y_{i,a}^{1-m} \frac{\partial y_n(\bar{Y}_a)}{\partial y_i}(-y_{i,a}^m) \tag{4.2.24}$$

By substituting Eq. (4.2.24) into Eq. (4.2.23a), the approximation becomes

$$\tilde{y}_n(\bar{Y}) = \beta + \frac{1}{m}\sum_{i=1}^{n-1} t_i y_{i,a}^{1-m} \frac{\partial y_n(\bar{Y}_a)}{\partial y_i} y_i^m \tag{4.2.25}$$

Eq. (4.2.25) has n unknown constants, m and t_i $(i = 1, 2, ..., n-1)$. They can be evaluated by using the following n equations, that is, by letting the approximate function value and the derivatives at $\bar{Y}_b$ equal their corresponding exact values at this point,

$$\frac{\partial y_n(\bar{Y}_b)}{\partial y_i} = t_i \left(\frac{y_{i,a}}{y_{i,b}}\right)^{1-m} \frac{\partial y_n(\bar{Y}_a)}{\partial y_i}, \qquad i = 1, 2, ..., n-1 \tag{4.2.26a}$$

$$\tilde{y}_n(\bar{Y}_b) = \beta + \frac{1}{m}\sum_{i=1}^{n-1} t_i y_{i,a}^{1-m} \frac{\partial y_n(\bar{Y}_a)}{\partial y_i} y_{i,b}^m \tag{4.2.26b}$$

where t_i $(i = 1, 2, ..., n-1)$ is a real number, which gives additional degrees of freedom for improving the approximation accuracy since the nonlinear index, m, is an integer. If they are equal to 1, Eq. (4.2.23) is a Taylor series expansion. Eq. (4.2.26) has n unknown constants, and they can be determined by using a simple adaptive search technique given in Section 3.3.3.6. The approximate function values at Y_a, Y_b and MPP, and derivatives at Y_b are equal to their respective exact values. The function is a paraboloid approximation when m is equal to 2. The approximate function can be used for the problems shown in Figs. 4.5a and 4.5b where the paraboloid approximations can't provide any improvement over the FORM. This approximation

is simple to implement with less computation even when the number of variables is large and can also be used for highly nonlinear problems.

The approximation given in Eq. (4.2.25) is rewritten as

$$\tilde{y}_n(\bar{Y}) = \beta + \sum_{i=1}^{n-1} a_i y_i^m \tag{4.2.27a}$$

where

$$a_i = t_i \frac{y_{i,a}^{1-m}}{m} \frac{\partial y_n(\bar{Y}_a)}{\partial y_i} \tag{4.2.27b}$$

4.3 First Order Reliability Method (FORM)

After approximating the response surface, the failure probability can be calculated by integrating Eq.(4.1.6a). In the first order reliability method, the limit state surface is approximated by the tangent plane at the MPP given in Eq. (4.2.9). Therefore, the approximate failure region Ω is defined as

$$\Omega = \{Y | y_n - \beta > 0\} \tag{4.3.1}$$

Since the random variables in Y-space are independent, standard and normally distributed, the n-dimensional standardized normal probability density function can be written as

$$\begin{aligned} \Psi(y_1, y_2, ..., y_n) &= \frac{1}{(2\pi)^{\frac{n}{2}}} exp[-\frac{1}{2}(y_1^2 + y_2^2 + ... + y_n^2)] \\ &= \phi(y_1)\phi(y_2)...\phi(y_n) \end{aligned} \tag{4.3.2}$$

where $\phi(y_i)$ is the probability density function for the ith random variable with a standard normal distribution.

The failure probability given in Eq. (4.1.6a) can be computed from a formulation in Y-space.

$$P_f = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(y_1) \ldots \phi(y_{n-1}) \int_{\beta}^{\infty} \phi(y_n) dy_n dy_{n-1} \ldots dy_1$$
$$= 1 - \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(y_1) \ldots \phi(y_{n-1}) \int_{-\infty}^{\beta} \phi(y_n) dy_n dy_{n-1} \ldots dy_1 \quad (4.3.3)$$

Because

$$\int_{-\infty}^{\beta} \phi(y_n) dy_n = \Phi(\beta) \quad (4.3.4)$$

and

$$\int_{-\infty}^{+\infty} \phi(y_i) dy_i = 1, \qquad i = 1, 2, \ldots, n-1 \quad (4.3.5)$$

Substituting these two equations into Eq. (4.3.3), we obtain

$$P_f = 1 - \Phi(\beta) = \Phi(-\beta) \quad (4.3.6)$$

Eq. (4.3.6) provides the exact estimate of the probability of failure if the limit state function is linear and the random variables are normal distributions. Since the approximation to the response surface is the first-order Taylor approximation, the method is called the first-order reliability method (FORM). FORM usually works well when the limit-state surface has only one minimal distance point and the function is nearly linear in the neighborhood of the design point. However, if the failure surface has large curvatures (high nonlinearity), the failure probability estimated using the safety index β by FORM may give unreasonable and inaccurate results [7] and more accurate approximate methods have to be applied.

Example 4.5

This example was given in Example 3.12 of Chapter 2. The performance function is

$$g(x_1, x_2) = x_1^4 + 2x_2^4 - 20$$

where x_1 and x_2 are the random variables with normal distribution (mean $\bar{x}_1 = x_2 = 10.0$, standard deviation $\sigma_1 = \sigma_2 = 5.0$). The safety index β was obtained from Example 3.12, *i.e.* $\beta = 2.3654$. Using Eq. (4.3.6) (FORM), the failure probability is computed as

$$P_f = \Phi(-\beta) = \Phi(-2.3654) = 0.009$$

The failure probability using the Monte Carlo method (sample size=100,000, seed=5000) is 0.001950. Compared with this result, the first-order reliability method is inaccurate for this highly nonlinear problem. Therefore, more accurate approximate methods are needed.

4.4 Second Order Reliability Method (SORM)

As mentioned in the previous section, if the failure surface has large curvatures, *i.e.*, high nonlinearity, the failure probability estimated using the safety index β by FORM may give unreasonable and inaccurate results (Figure 4.6) [7]. For the linear limit state *bb*, containing P_1 as design point, the failure probability for normal variables is given exactly by $P_f = \Phi(-\beta)$. However, the point P_1 is also the design point for nonlinear limit state functions *aa* and *cc*. In terms of first-order reliability theory, each of these limit states has an identical value of β, and hence an identical nominal failure probability $P_f = \Phi(-\beta)$; however it is quite clear from Figure 4.6 that the actual probability contents of the respective failure regions are not identical. Similarly, the limit state *dd* represents probably a lower failure probability still; yet its safety margin β_1 is less than β. Evidently β as defined so far lacks a sense of "comparativeness" or an "ordering property" with respect to the implied probability content for nonlinear limit states.

A further point of interest is that no limitation has been placed on the direction of β in U-space so that, for some other checking point P_2, the probability content for the linear limit state *ee* should be identical with that implied by *bb* when both have the same distance β from the origin.

In the above cases, more accurate approximate methods have to be applied. Breitung [8], Tvedt [9][10], Hohenbichler and Rackwitz [11], Koyluoglu and Nielsen [12], and Cai and Elishakoff [13] have developed second-order reliability methods (SORM) using the second-order approximation given in Eq. (4.2.19) to replace the original failure surfaces. Wang and Grandhi

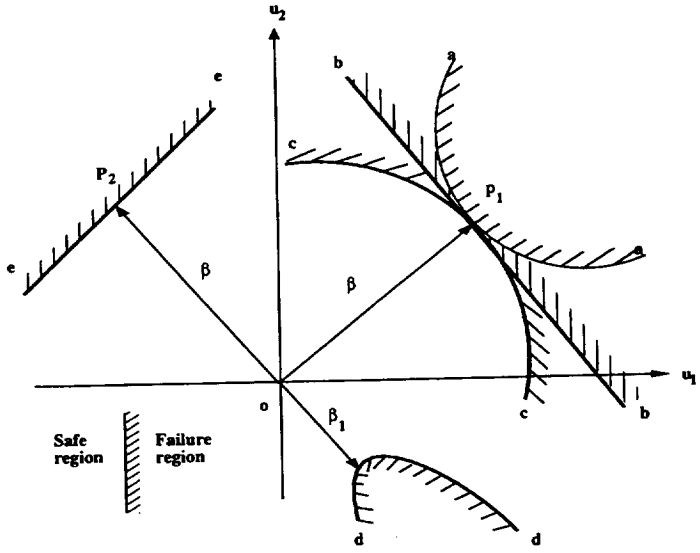


Fig. 4.6 Inconsistency between β and P_{fN} for different forms of limit state functions

[3] and Der Kiureghian, *et al* [4] calculated second-order failure probabilities using approximate curvatures to avoid exact second-order derivatives calculations of the limit state surface.

Breitung's and Tvedt's formulations are introduced in Sections 4.4.1 and 4.4.2, respectively. Wang and Grandhi's SORM with approximate curvatures calculations is given in Section 4.4.3.

4.4.1 Breitung's Formulation

To explain the Breitung formulation, first, a Laplace method for the asymptotic approximation of multidimensional integrals is needed. Defining

$$I(\beta) = \int_{g(Y)<0} exp(\frac{-\beta^2|Y|^2}{2})dY \tag{4.3.7}$$

where $I(\beta)$ is an integral over a fixed domain whose integrand is an exponential function depending linearly on the parameter β^2. An extensive study of the asymptotic behavior for β^2 is described in Ref.[14], Chapt. 8. Using the results given there, the asymptotic form of $I(\beta)$ is (details see Appendix D)

$$I(\beta) \sim (2\pi)^{(n-1)/2} exp(-\frac{\beta^2}{2}) \beta^{-(n+1)} |J|^{-1/2}, \quad \beta \to \infty \tag{4.3.8}$$

where J is a quantity independent of β depending only on the first and second derivatives of the failure surface at the MPP, which is defined in Eq. (D.3) in Appendix D.

In the case of independent standard normal random variables, the joint probability density function (PDF) is given by Eq. (4.3.2), *i.e.*,

$$P_f = (2\pi)^{-n/2} \int_{g(U)<0} exp(-\frac{|U|^2}{2}) dU \tag{4.3.9}$$

Substituting $(x_1, x_2, ..., x_n) \to (y_1, y_2, ..., y_n)$ with $y_i = \beta^{-1} u_i$:

$$P_f = (2\pi)^{-n/2} \beta^n \int_{g(Y)<0} exp(-\beta^2 \frac{|Y|^2}{2}) dY \tag{4.3.10}$$

Substituting Eq. (4.3.8) into this equation, we obtain

$$P_f \sim (2\pi)^{-1/2} \beta^{-1} exp(-\frac{\beta^2}{2}) |J|^{-1/2}, \qquad \beta \to \infty \tag{4.3.11}$$

Since the failure surface is approximated by the quadratic Taylor series expansion at the MPP given in Eq. (4.2.12), $|J|$ can be computed based on Eq. (D.9) given in Appendix D,

$$|J| = \sum_{i=1}^{p} |J_i| = \sum_{i=1}^{n} \prod_{j=1}^{n-1} (1 + \kappa_{ij} \beta) \tag{4.3.12}$$

where p is the number of points on $g(U) = 0$ with the shortest distance β from the origin to the failure surface, and κ_{ij} is the main curvature of the failure surface at the MPP. If there is

only one MPP on the surface, by substituting this equation into Eq. (4.3.11) and considering Mill's ratio

$$\Phi(-Y) \sim (2\pi)^{-1/2} Y^{-1} exp(-Y^2/2) \tag{4.3.13}$$

P_f can be computed

$$P_f \approx \Phi(-\beta) \prod_{j=1}^{n-1} (1 + \kappa_j \beta)^{-1/2} \tag{4.3.14}$$

Since Eq. (4.3.14)is an analytical equation, it is easy to implement the Breitung algorithm. The main steps of Breitung's formulation include:

1). Conducting the safety index search and locating the MPP, U^*;

2). Computing the second-order derivatives of the limit state surface at U^* and forming the B matrix given in Eq. (4.2.12c);

3). Calculating the orthogonal matrix H based on the procedure given in Section 4.2.1;

4). Computing the main curvatures κ_j of the failure surface at the MPP using Eq. (4.2.20);

5). Computing the failure probability P_f using Eq. (4.3.14).

Example 4.6

Compute the failure probability P_f using the Breitung method (Eq. 4.3.14) for Example 4.5.

1). Compute the safety index and MPP, U^*;

The safety index β was calculated as $\beta = 2.3654$ from Example 3.12. The MPP was located at $U^*(-1.6368, -1.7077)$ (in X-space, $X^*(1.8162, 1.4613)$).

2). Compute the second-order derivatives of the limit state surface at U^* and form the B matrix given in Eq. (4.2.12c);

$$\frac{\partial g}{\partial u_1} = \frac{\partial g}{\partial x_1}\sigma_1 = 4x_1^3\sigma_1 = 4 \times 1.8162^3 \times 5 = 119.8148$$
$$\frac{\partial g}{\partial u_2} = \frac{\partial g}{\partial x_2}\sigma_1 = 4x_2^3\sigma_2 = 8 \times 1.4613^3 \times 5 = 124.8218$$

$$\frac{\partial^2 g}{\partial x_1^2} = 12x_1^2 = 12 \times 1.8162^2 = 39.583$$

$$\frac{\partial^2 g}{\partial x_2^2} = 24x_2^2 = 24 \times 1.4613^2 = 51.2495$$

$$\frac{\partial^2 g}{\partial u_1^2} = \frac{\partial^2 g}{\partial x_1^2}\sigma_1^2 = 39.583 \times 5^2 = 989.5592$$

$$\frac{\partial^2 g}{\partial u_2^2} = \frac{\partial^2 g}{\partial x_2^2}\sigma_2^2 = 51.2495 \times 5^2 = 1281.2632$$

$$\begin{aligned} |\nabla g(U^*)| &= \sqrt{(\frac{\partial g}{\partial u_1})^2 + (\frac{\partial g}{\partial u_2})^2} \\ &= \sqrt{(119.8148)^2 + (124.8218)^2} \\ &= 173.0205 \end{aligned}$$

$$\begin{aligned} B &= \frac{\nabla^2 g(U^*)}{|\nabla g(U^*)|} \\ &= \frac{1}{|\nabla g(U^*)|} \begin{pmatrix} 989.5592 & 0 \\ 0 & 1281.2632 \end{pmatrix} \end{aligned}$$

3). Calculate the orthogonal matrix H based on the procedure given in Section 4.2.1;

$$H = \begin{pmatrix} \gamma_2{}^T \\ \gamma_1{}^T \end{pmatrix} = \begin{pmatrix} -0.7214 & 0.6925 \\ -0.6924 & -0.7214 \end{pmatrix}$$

4). Compute the main curvatures κ_j by solving the eigenvalues of HBH^T;

$$HBH^T = \begin{pmatrix} 6.5278 & -0.8423 \\ -0.8423 & 6.5968 \end{pmatrix}$$

An eigenvalue of the above matrix is solved as $\kappa_1 = 6.5278$, so the main curvature of the failure surface at the MPP is 6.5278.

5). Compute the failure probability P_f using the Breitung formula

$$
\begin{aligned}
P_f &= \Phi(-\beta)\prod_{j=1}^{n-1}(1+\kappa_j\beta)^{-1/2} \\
&= \Phi(-2.3654)(1+6.5278\times 2.3654)^{-1/2} \\
&= 9.0040\times 10^{-3}\times 0.2466 \\
&= 0.00222059
\end{aligned}
$$

Compared with the FORM's result ($P_f = 0.009004$), Breitung's method is closer to the Monte Carlo result ($P_f = 0.001950$). It generally provides better results than FORM due to the second-order approximation. However, this method is not valid for $\beta\kappa_j \leq -1$ and does not work well in the case of negative curvatures.

4.4.2 Tvedt's Formulation

Based on the second-order approximation of the failure surface given in Eq. (4.2.19), the approximate failure region Ω is defined as

$$\Omega = \{Y | y_n - (\beta + \frac{1}{2}\sum_{i=1}^{n-1}\kappa_i y_i'^2) > 0\} \tag{4.3.15}$$

The failure probability given in Eq. (4.1.6a) can be computed from a formulation in Y-space.

$$P_f = 1 - \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\phi(y_1)\ldots\phi(y_{n-1})\int_{\beta+\frac{1}{2}\sum_{i=1}^{n-1}\kappa_i y_i'^2}^{\infty}\phi(y_n)dy_n dy_{n-1}\ldots dy_1 \tag{4.3.16}$$

Tvedt has derived a three-term approximation to this equation by a power series expansion in terms of $\frac{1}{2}\sum_{i=1}^{n-1}\kappa_i y_i'^2$, ignoring terms of order higher than two. The resultant approximation for P_f is

$$A_1 = \Phi(-\beta)\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2} \tag{4.3.17a}$$

$$A_2 = [\beta\Phi(-\beta) - \phi(\beta)]\cdot\{\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2} - \prod_{i=1}^{n-1}(1+(\beta+1)\kappa_i)^{-1/2}\} \tag{4.3.17b}$$

$$A_3 = (\beta+1)[\beta\Phi(-\beta) - \phi(\beta)] \cdot \{\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2} - Re\{\prod_{i=1}^{n-1}(1+(\beta+1)\kappa_i)^{-1/2}\} \quad (4.3.17c)$$

$$p_f = A_1 + A_2 + A_3 \quad (4.3.17d)$$

The first term, A_1 is the Breitung formula of Eq. (4.3.14). $Re[.]$ denotes the real part. This method has been found to give very good approximations in most cases. The asymptotic behavior of the three terms can be compared in the asymptotic sense used in Eq. (4.3.14). It may be shown that the ratio of the second term to the first term is

$$\frac{A_2}{A_1} \sim \frac{1}{2\beta^2}\sum_{j=1}^{n-1}\frac{\beta\kappa_j}{1-\beta\kappa_j}, \qquad \beta \to \infty \quad (4.3.18)$$

Similarly, the ratio of the third to the first term is

$$\frac{A_3}{A_1} \sim -\frac{3}{8\beta^2}\sum_{j=1}^{n-1}(\frac{\beta\kappa_j}{1-\beta\kappa_j})^2 - \frac{1}{2\beta^2}\sum_{j=1}^{n-1}\sum_{m=j+1}^{n-1}\frac{\beta^2\kappa_j\kappa_m}{(1-\beta\kappa_j)(1-\beta\kappa_m)}, \qquad \beta \to \infty \quad (4.3.19)$$

Since Eq. (4.3.19) is an analytical equation, it is easy to implement the algorithm. The main steps of the Tvedt's formulation are the same as Breitung's except step (5), *i.e.*, the failure probability P_f is calculated using Eq. (4.3.17).

Example 4.7

Compute the failure probability P_f using the Tvedt method (Eq. 4.3.17) for Example 4.5. The first four steps are the same as Example 4.6, i.e.,

1). Compute the safety index and MPP, U^*;

The safety index β was calculated as $\beta = 2.3654$ from Example 3.12. The MPP was located at $U^*(-1.6368, -1.7077)$ (in X-space, $X^*(1.8162, 1.4613)$).

2). Compute the second-order derivatives of the limit state surface at U^* and form the B matrix given in Eq. (4.2.12c);

$$\frac{\partial g}{\partial u_1} = \frac{\partial g}{\partial x_1}\sigma_1 = 4x_1^3\sigma_1 = 4 \times 1.8162^3 \times 5 = 119.8148$$

$$\frac{\partial g}{\partial u_2} = \frac{\partial g}{\partial x_2}\sigma_1 = 4x_2^3\sigma_2 = 8 \times 1.4613^3 \times 5 = 124.8218$$

$$\frac{\partial^2 g}{\partial x_1^2} = 12x_1^2 = 12 \times 1.8162^2 = 39.583$$

$$\frac{\partial^2 g}{\partial x_2^2} = 24x_2^2 = 24 \times 1.4613^2 = 51.2495$$

$$\frac{\partial^2 g}{\partial u_1^2} = \frac{\partial^2 g}{\partial x_1^2}\sigma_1^2 = 39.583 \times 5^2 = 989.5592$$

$$\frac{\partial^2 g}{\partial u_2^2} = \frac{\partial^2 g}{\partial x_2^2}\sigma_2^2 = 51.2495 \times 5^2 = 1281.2632$$

$$\begin{aligned} |\nabla g(U^*)| &= \sqrt{(\frac{\partial g}{\partial u_1})^2 + (\frac{\partial g}{\partial u_2})^2} \\ &= \sqrt{(119.8148)^2 + (124.8218)^2} \\ &= 173.0205 \end{aligned}$$

$$\begin{aligned} B &= \frac{\nabla^2 g(U^*)}{|\nabla g(U^*)|} \\ &= \frac{1}{|\nabla g(U^*)|}\begin{pmatrix} 989.5592 & 0 \\ 0 & 1281.2632 \end{pmatrix} \end{aligned}$$

3). Calculate the orthogonal matrix H based on the procedure given in Section 4.2.1;

$$H = \begin{pmatrix} \gamma_2 \\ \gamma_1 \end{pmatrix} = \begin{pmatrix} -0.7214 & 0.6925 \\ -0.6924 & -0.7214 \end{pmatrix}$$

4). Compute the main curvatures κ_j by solving the eigenvalues of HBH^T;

$$HBH^T = \begin{pmatrix} 6.5278 & -0.8423 \\ -0.8423 & 6.5968 \end{pmatrix}$$

An eigenvalue of the above matrix is solved as $\kappa_1 = 6.5278$, so the main curvature of the failure surface at the MPP is 6.5278.

5). Compute the failure probability P_f using the Tvedt formula (Eq. 4.3.17)

The first term of the Tvedt formula is the same as Breitung's method, so

$$A_1 = 0.00222059$$

$$\begin{aligned} A_2 &= [\beta\Phi(-\beta) - \phi(\beta)] \cdot \{\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2} - \prod_{i=1}^{n-1}(1+(\beta+1)\kappa_i)^{-1/2}\} \\ &= [2.3654 \times \Phi(-2.3654) - \phi(2.3654)] \\ &\quad \cdot\{(1+2.3654\times 6.5278)^{-1/2} - (1+(2.3654+1)\times 6.5278)^{-1/2}\} \\ &= 2.2205 \times 10^{-3} \end{aligned}$$

$$\begin{aligned} A_3 &= (\beta+1)[\beta\Phi(-\beta) - \phi(\beta)] \cdot \{\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2} - Re[\prod_{i=1}^{n-1}(1+(\beta+1)\kappa_i)^{-1/2}]\} \\ &= (2.3654+1)[2.3654 \times \Phi(-2.3654) - \phi(2.3654)] \\ &\quad \cdot\{(1+2.3654\times 6.5278)^{-1/2} - Re[(1+(2.3654+1)\times 6.5278)^{-1/2}]\} \\ &= -1.3297 \times 10^{-4} \end{aligned}$$

$$P_f = A_1 + A_2 + A_3 = 0.00222059 + 2.2205 \times 10^{-3} - 1.3297 \times 10^{-4} = 0.002087$$

Compared to the FORM result ($P_f = 0.009004$) and the Breitung result ($P_f = 0.00222059$), Tvedt's method is closer to the Monte Carlo ($P_f = 0.001950$).

Like Breitung's algorithm, Tvedt's method is also invalid for $\beta\kappa_j \leq -1$ and does not work well in the case of negative curvatures.

4.4.3 SORM with Approximate Curvatures

In this method, Breitung's and Tvedt's formulas are used to perform the failure probability calculations. However, the main curvatures are approximately calculated by using the nonlinear approximation constructed during the safety index search of this work.

The main steps of the SORM having approximate curvatures are summarized as follows for a complete failure probability analysis.

1). Conducting the safety index search and locating the MPP, U^*;

2). Computing the second-order derivatives of the limit state surface at U^* using Eq. (4.2.21) and forming the B matrix given in Eq. (4.2.12c);

3). Calculating the orthogonal matrix H based on the procedure given in Section 4.2.1;

4). Computing the approximate curvatures κ_j of the failure surface at the MPP using Eq.(4.2.20);

5). Computing the failure probability P_f using Breitung's formula of Eq. (4.3.14) or Tvedt's formula of Eq. (4.3.17).

A significant reduction in computer effort comes from the use of approximate functions in step 2 for the second-order derivatives because exact analysis is avoided. Therefore this method is particularly suitable for problems having implicit performance functions needing large scale finite element models for structural analysis.

Example 4.8

Compute the failure probability P_f using Wang and Grandhi's SORM for Example 4.5.

1). Compute the safety index and the MPP, U^*;

The safety index β was calculated as $\beta = 2.3654$ from Example 3.12. The MPP was located at $U^*(-1.6368, -1.7077)$ (in X-space, $X^*(1.8162, 1.4613)$).

2). Compute approximate second-order derivatives at U^* using Eq. (4.2.21) and form the B matrix given in Eq. (4.2.12c);

$$\frac{\partial g}{\partial u_1} = \frac{\partial g}{\partial x_1}\sigma_1 = 4x_1^3\sigma_1 = 4 \times 1.8162^3 \times 5 = 119.8148$$

$$\frac{\partial g}{\partial u_2} = \frac{\partial g}{\partial x_2}\sigma_1 = 4x_2^3\sigma_2 = 8 \times 1.4613^3 \times 5 = 124.8218$$

$$\frac{\partial^2 \tilde{g}}{\partial u_1^2} = 989.5592$$

$$\frac{\partial^2 g}{\partial u_2^2} = 1281.2631$$

$$\begin{aligned} |\nabla g(U^*)| &= \sqrt{(\frac{\partial g}{\partial u_1})^2 + (\frac{\partial g}{\partial u_2})^2} \\ &= \sqrt{(119.8148)^2 + (124.8218)^2} \\ &= 173.0205 \end{aligned}$$

$$\begin{aligned} B &= \frac{\nabla^2 g(U^*)}{|\nabla g(U^*)|} \\ &= \frac{1}{|\nabla g(U^*)|} \begin{pmatrix} 989.5592 & 0 \\ 0 & 1281.2631 \end{pmatrix} \end{aligned}$$

3). Calculate the orthogonal matrix H based on the procedure given in Section 4.2.1;

$$H = \begin{pmatrix} \gamma_2 \\ \gamma_1 \end{pmatrix} = \begin{pmatrix} -0.7214 & 0.6925 \\ -0.6924 & -0.7214 \end{pmatrix}$$

4). Compute the main curvatures κ_j by solving the eigenvalues of HBH^T;

$$HBH^T = \begin{pmatrix} 6.5278 & -0.8423 \\ -0.8423 & 6.5968 \end{pmatrix}$$

5). Compute the failure probability P_f using the Tvedt formula (Eq. 4.3.17)

The first term of the Tvedt formula is the same as Breitung's method, so

$$A_1 = 0.00222059$$

$$\begin{aligned} A_2 &= [\beta\Phi(-\beta) - \phi(\beta)] \cdot \{\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2} - \prod_{i=1}^{n-1}(1+(\beta+1)\kappa_i)^{-1/2}\} \\ &= [2.3654 \times \Phi(-2.3654) - \phi(2.3654)] \\ &\quad \cdot\{(1+2.3654\times 6.5278)^{-1/2} - (1+(2.3654+1)\times 6.5278)^{-1/2}\} \\ &= 2.2206 \times 10^{-3} \end{aligned}$$

$$
\begin{aligned}
A_3 &= (\beta+1)[\beta\Phi(-\beta)-\phi(\beta)]\cdot\{\prod_{i=1}^{n-1}(1+\beta\kappa_i)^{-1/2}-Re[\prod_{i=1}^{n-1}(1+(\beta+1)\kappa_i)^{-1/2}]\} \\
&= (2.3654+1)[2.3654\times\Phi(-2.3654)-\phi(2.3654)] \\
&\quad\cdot\{(1+2.3654\times 6.5278)^{-1/2}-Re[(1+(2.3654+1)\times 6.5278)^{-1/2}]\} \\
&= -1.3298\times 10^{-4}
\end{aligned}
$$

$$
P_f = A_1 + A_2 + A_3 = 0.00222059 + 2.2206\times 10^{-3} - 1.3298\times 10^{-4} = 0.002088
$$

This result is very close to the Tvedt's result ($P_f = 0.002087$) with the exact second-order gradients of the limit state surface. It means that the approximation given in Eq. (4.2.21) accurately represents the real failure surface in this example. Since this method doesn't require any exact second-order gradient calculations, it can be used for problems where the second-order gradients are expensive or impossible to calculate.

4.5 Higher Order Reliability Method (HORM)

Based on the higher-order approximation of the failure surface given in Eq. (4.2.27), the approximate failure region Ω is defined as

$$
\Omega = \{Y | y_n - (\beta + \sum_{i=1}^{n-1} a_i y_i^m) > 0\} \tag{4.3.20}
$$

Since the random variables in Y-space are independent, standard and normally distributed, the n-dimensional standardized normal probability density function can be written as

$$
\begin{aligned}
\Psi(y_1, y_2, ..., y_n) &= \frac{1}{(2\pi)^{\frac{n}{2}}} exp[-\frac{1}{2}(y_1^2 + y_2^2 + ... + y_n^2)] \\
&= \phi(y_1)\phi(y_2)...\phi(y_n)
\end{aligned} \tag{4.3.21}
$$

where $\phi(y_i)$ is the probability density function for the ith random variable with a standard normal distribution.

The failure probability given in Eq. (4.1.6a) can be computed from a formulation in Y-space.

$$P_f = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(y_1) \ldots \phi(y_{n-1}) \int_{\beta + \sum_{i=1}^{n-1} a_i y_i^m}^{\infty} \phi(y_n) dy_n dy_{n-1} \ldots dy_1$$

$$= 1 - \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(y_1) \ldots \phi(y_{n-1}) \int_{-\infty}^{\beta + \sum_{i=1}^{n-1} a_i y_i^m} \phi(y_n) dy_n dy_{n-1} \ldots dy_1 \quad (4.3.22)$$

Dividing the innermost integration of the second term in Eq. (4.3.22) over the interval $(-\infty, \beta + \sum_{i=1}^{n-1} a_i y_i^m]$ into integrals over $(-\infty, \beta]$ and $(\beta, \beta + \sum_{i=1}^{n-1} a_i y_i^m]$, Eq. (4.3.22) becomes

$$P_f = 1 - \Phi(\beta) - \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(y_1) \ldots \phi(y_{n-1}) \int_{\beta}^{\beta + \sum_{i=1}^{n-1} a_i y_i^m} \phi(y_n) dy_n dy_{n-1} \ldots dy_1$$

$$= \Phi(-\beta) - \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi(y_1) \ldots \phi(y_{n-1}) [\Phi(\beta + V) - \Phi(\beta)] dy_{n-1} \ldots dy_1 \quad (4.3.23)$$

where

$$V = \sum_{i=1}^{n-1} a_i y_i^m \quad (4.3.24)$$

The sign of V depends on the sign of a_i and y_i^m. The failure probability calculations for the following five cases are described below.

4.5.1 Case 1 - All a_i are positive and m is even

If all a_i are positive and m is even, V is positive. Let $f(V)$ represent the integral function given in Eq. (4.3.23), that is

$$f(V) = \Phi(\beta + V) - \Phi(\beta) \quad (4.3.25)$$

The following expansion function of $f(V)$ given in Ref. [12] is used, which is

$$\Phi(\beta+V) - \Phi(\beta) = (1 - \Phi(\beta)) \{1 - exp(-\frac{V}{c_{0,1}})(1 + c_{1,1}V + c_{2,1}V^2 + c_{3,1}V^3 + \ldots + c_{N,1}V^N)\} \quad (4.3.26)$$

where N is the number of the expansion terms. This function fulfills

$$f(0) = 0 \tag{4.3.27a}$$

$$\lim_{V\to\infty} f(V) = 1 - \Phi(\beta) \tag{4.3.27b}$$

Based on the expansion function of Eq. (4.3.26), the failure probability is computed as

$$\begin{aligned} P_f \;=\; & \frac{\Phi(-\beta)}{(2\pi)^{\frac{n-1}{2}}} \prod_{j=1}^{n-1} H_j^{(0)} \{1 + c_{1,1} \sum_{k=1}^{n-1} \frac{H_k^{(1)}}{H_k^{(0)}} + c_{2,1}[\sum_{k=1}^{n-1} \frac{H_k^{(2)}}{H_k^{(0)}} + \sum_{k=1}^{n-1} \sum_{l=1,l\neq k}^{n-1} \frac{H_k^{(1)} H_l^{(1)}}{H_k^{(0)} H_l^{(0)}}] \\ & + c_{3,1}[\sum_{k=1}^{n-1} \frac{H_k^{(3)}}{H_k^{(0)}} + 3 \sum_{k=1}^{n-1} \sum_{l=1,l\neq k}^{n-1} \frac{H_k^{(2)} H_l^{(1)}}{H_k^{(0)} H_l^{(0)}} + \sum_{k=1}^{n-1} \sum_{l=1,l\neq k}^{n-1} \sum_{p=1,p\neq l,p\neq k}^{n-1} H_k^{(1)} H_l^{(1)} H_p^{(1)} \end{aligned} \tag{4.3.28}$$

where

$$H^{(0)} = \int_{-\infty}^{\infty} e^{-\frac{1}{2}y^2 - \frac{a}{c_{0,1}} y^m} dy \tag{4.3.29a}$$

$$H^{(1)} = \int_{-\infty}^{\infty} a y^m e^{-\frac{1}{2}y^2 - \frac{a}{c_{0,1}} y^m} dy \tag{4.3.29b}$$

$$H^{(2)} = \int_{-\infty}^{\infty} a^2 y^{2m} e^{-\frac{1}{2}y^2 - \frac{a}{c_{0,1}} y^m} dy \tag{4.3.29c}$$

$$H^{(3)} = \int_{-\infty}^{\infty} a^3 y^{3m} e^{-\frac{1}{2}y^2 - \frac{a}{c_{0,1}} y^m} dy \tag{4.3.29d}$$

By using the ten-term Hermite approximate integral formula, the integration of Eq. (4.3.29) can be computed as

$$H^{(0)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}} \xi_i^m} \tag{4.3.30a}$$

$$H^{(1)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a\xi_i^m \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}} \xi_i^m} \tag{4.3.30b}$$

$$H^{(2)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^2 \xi_i^{2m} \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}} \xi_i^m} \tag{4.3.30c}$$

$$H^{(3)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^3 \xi_i^{3m} \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}} \xi_i^m} \tag{4.3.30d}$$

where λ_i and ξ_i are Hermite integral parameters which are given in Table E1 of Appendix E.

If $m = 2$, the integration of Eq. (4.3.29) can be analytically integrated. The analytical formula for Case 1 was derived in Ref. [12], that is

$$\begin{aligned} P_f &= \Phi(-\beta)\prod_{j=1}^{n-1}\frac{1}{\sqrt{1+a_j/c_{0,1}}} \\ &\cdot\{1+\frac{1}{2}c_{1,1}\sum_{i=1}^{n-1}\frac{a_i}{1+a_i/c_{0,1}}+\frac{1}{4}c_{2,1}[(\sum_{i=1}^{n-1}\frac{a_i}{1+a_i/c_{0,1}})^2+2\sum_{i=1}^{n-1}(\frac{a_i}{1+a_i/c_{0,1}})^2] \\ &+\frac{1}{8}c_{3,1}\{(\sum_{i=1}^{n-1}\frac{a_i}{1+a_i/c_{0,1}})^3+2\sum_{i=1}^{n-1}\frac{a_i}{1+a_i/c_{0,1}}\sum_{i=1}^{n-1}(\frac{a_i}{1+a_i/c_{0,1}})^2 \\ &+12\sum_{i=1}^{n-1}(\frac{a_i}{1+a_i/c_{0,1}})^3\}+...\} \end{aligned} \tag{4.3.31}$$

4.5.2 Case 2 - All a_i are negative and m is even

If all a_i are negative and m is even, V is negative. The following expansion function of $f(V)$ given in Ref. [12] is used, *i.e.*

$$\Phi(\beta+V)-\Phi(\beta)=-\Phi(\beta)\{1-exp(\frac{V}{c_{0,2}})(1+c_{1,2}V+c_{2,2}V^2+c_{3,2}V^3+...+c_{N,2}V^N)\} \tag{4.3.32}$$

which fulfills

$$f(0)=0 \tag{4.3.33a}$$

$$\lim_{V\to-\infty} f(V)=-\Phi(\beta) \tag{4.3.33b}$$

Based on Eq. (23), the failure probability is computed as

$$\begin{aligned} P_f &= 1-\frac{\Phi(\beta)}{(2\pi)^{\frac{n-1}{2}}}\prod_{j=1}^{n-1}\bar{H}_j^{(0)}\{1+c_{1,2}\sum_{k=1}^{n-1}\frac{\bar{H}_k^{(1)}}{\bar{H}_k^{(0)}}+c_{2,2}[\sum_{k=1}^{n-1}\frac{\bar{H}_k^{(2)}}{\bar{H}_k^{(0)}}+\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\frac{\bar{H}_k^{(1)}\bar{H}_l^{(1)}}{\bar{H}_k^{(0)}\bar{H}_l^{(0)}}] \\ &+c_{3,2}[\sum_{k=1}^{n-1}\frac{\bar{H}_k^{(3)}}{\bar{H}_k^{(0)}}+3\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\frac{\bar{H}_k^{(2)}\bar{H}_l^{(1)}}{\bar{H}_k^{(0)}\bar{H}_l^{(0)}}+\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\sum_{p=1,p\neq l,p\neq k}^{n-1}\frac{\bar{H}_k^{(1)}\bar{H}_l^{(1)}\bar{H}_p^{(1)}}{\bar{H}_k^{(0)}\bar{H}_l^{(0)}\bar{H}_p^{(0)}}]+...\} \end{aligned}$$

(4.3.34)

where

$$\bar{H}^{(0)} = \int_{-\infty}^{\infty} e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}\,dy \tag{4.3.35a}$$

$$\bar{H}^{(1)} = \int_{-\infty}^{\infty} a y^m e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}\,dy \tag{4.3.35b}$$

$$\bar{H}^{(2)} = \int_{-\infty}^{\infty} a^2 y^{2m} e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}\,dy \tag{4.3.35c}$$

$$\bar{H}^{(3)} = \int_{-\infty}^{\infty} a^3 y^{3m} e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}\,dy \tag{4.3.35d}$$

By using the ten-term Hermite approximate integral formula, the integration of Eq. (4.3.35) can be computed as

$$\bar{H}^{(0)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot e^{-\frac{1}{2}\xi_i^2+\frac{a}{c_{0,2}}\xi_i^m} \tag{4.3.36a}$$

$$\bar{H}^{(1)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a\xi_i^m \cdot e^{-\frac{1}{2}\xi_i^2+\frac{a}{c_{0,2}}\xi_i^m} \tag{4.3.36b}$$

$$\bar{H}^{(2)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^2\xi_i^{2m} \cdot e^{-\frac{1}{2}\xi_i^2+\frac{a}{c_{0,2}}\xi_i^m} \tag{4.3.36c}$$

$$\bar{H}^{(3)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^3\xi_i^{3m} \cdot e^{-\frac{1}{2}\xi_i^2+\frac{a}{c_{0,2}}\xi_i^m} \tag{4.3.36d}$$

If $m = 2$, Eq. (4.3.35) can be analytically integrated. The analytical formula for this case was derived in Ref. [12] that is

$$\begin{aligned} P_f \;=\; & 1-\Phi(\beta)\prod_{j=1}^{n-1}\frac{1}{\sqrt{1-a_j/c_{0,2}}} \\ & \cdot\{1+\frac{1}{2}c_{1,2}\sum_{i=1}^{n-1}\frac{a_i}{1-a_i/c_{0,2}}+\frac{1}{4}c_{2,2}[(\sum_{i=1}^{n-1}\frac{a_i}{1-a_i/c_{0,2}})^2+2\sum_{i=1}^{n-1}(\frac{a_i}{1-a_i/c_{0,2}})^2] \\ & +\frac{1}{8}c_{3,2}[(\sum_{i=1}^{n-1}\frac{a_i}{1-a_i/c_{0,2}})^3+2\sum_{i=1}^{n-1}\frac{a_i}{1-a_i/c_{0,2}}\sum_{i=1}^{n-1}(\frac{a_i}{1-a_i/c_{0,2}})^2+12\sum_{i=1}^{n-1}(\frac{a_i}{1-a_i/c_{0,2}})^3]+\ldots\} \end{aligned} \tag{4.3.37}$$

4.5.3 Case 3 - Some a_i are positive, some a_i are negative, and m is even

If some a_i are positive, some a_i are negative and m is even, V is divided as

$$V = V_1 + V_2 \tag{4.3.38}$$

where

$$V_1 = \sum_{i=1}^{\bar{n}-1} a_i y_i^m > 0, \qquad a_i > 0 \quad (i = 1, 2, ..., \bar{n}-1) \tag{4.3.39a}$$

$$V_2 = \sum_{i=\bar{n}}^{n-1} a_i y_i^m < 0, \qquad a_i < 0 \quad (i = \bar{n}, \bar{n}+1, ..., n-1) \tag{4.3.39b}$$

The following expansion function of $f(V)$ given in Ref. [12] is used, *i.e.*

$$\begin{aligned}\Phi(\beta + V_1 + V_2) - \Phi(\beta) \;=\; & \Phi(-\beta)\{1 - exp(-\frac{V_1}{c_{0,1}})(1 + c_{1,1}V_1 + c_{2,1}V_1^2 + ...)\} \\ & \cdot exp(\frac{V_2}{d_{0,2}})(1 + d_{1,2}V_2 + ...) \\ & -\Phi(\beta)\{1 - exp(\frac{V_2}{c_{0,2}})(1 + c_{1,2}V_2 + c_{2,2}V_2^2 + ...)\} \\ & \cdot exp(-\frac{V_1}{d_{0,1}})(1 + d_{1,1}V_1 + ...)\end{aligned} \tag{4.3.40}$$

This expansion function satisfies

$$f(0) = 0, \qquad V_1 = V_2 = 0 \tag{4.3.41a}$$

$$\lim_{V_1 \to \infty} f(V) = 1 - \Phi(\beta), \qquad at \quad V_2 = 0 \tag{4.3.41b}$$

$$\lim_{V_2 \to -\infty} f(V) = -\Phi(\beta), \qquad at \quad V_1 = 0 \tag{4.3.41c}$$

Based on Eq. (4.3.40), the failure probability can be computed as

$$\begin{aligned}P_f \;=\; & \Phi(-\beta) - \frac{\Phi(-\beta)}{(2\pi)^{\frac{n-\bar{n}}{2}}} \prod_{j=\bar{n}}^{n-1} \bar{H}_j^{(0)} (1 + d_{1,2} \sum_{k=\bar{n}}^{n-1} \frac{\bar{H}_k^{(1)}}{\bar{H}_k^{(0)}} + ...) \\ & \cdot \{1 - \prod_{j=1}^{\bar{n}-1} H_j^{(0)} (1 + c_{1,1} \sum_{k=1}^{\bar{n}-1} \frac{H_k^{(1)}}{H_k^{(0)}} + ...) - ...\}\end{aligned}$$

$$+\frac{\Phi(\beta)}{(2\pi)^{\frac{n-1}{2}}}\prod_{j=1}^{\bar{n}-1}H_j^{(0)}(1+d_{1,1}\sum_{k=1}^{\bar{n}-1}\frac{H_k^{(1)}}{H_k^{(0)}}+...)$$
$$\cdot\{1-\prod_{j=\bar{n}}^{n-1}\bar{H}_j^{(0)}(1+c_{1,2}\sum_{k=\bar{n}}^{n-1}\frac{\bar{H}_k^{(1)}}{\bar{H}_k^{(0)}}+...)-...\} \tag{4.3.42}$$

where $H^{(i)}(i=0,1,2,3)$ can be calculated using Eq. (4.3.29) or (4.3.30), but $c_{0,1}$ is replaced by $d_{0,1}$. Also, $\bar{H}^{(i)}(i=0,1,2,3)$ can be computed using Eq. (4.3.35) or (4.3.36), and $c_{0,2}$ is replaced by $d_{0,2}$.

If $m=2$, Eq. (4.3.42) can be analytically integrated, and the analytical formula for this case was derived in Ref. [12], that is

$$\begin{aligned}P_f &= \Phi(-\beta)-\Phi(-\beta)\prod_{j=\bar{n}}^{n-1}\frac{1}{\sqrt{1-a_j/d_{0,2}}}(1+\frac{1}{2}d_{1,2}\sum_{i=\bar{n}}^{n-1}\frac{a_i}{1-a_i/d_{0,2}}+...)\\
&\cdot\{1-\prod_{j=1}^{\bar{n}-1}\frac{1}{\sqrt{1+a_j/c_{0,1}}}(1+\frac{1}{2}c_{1,1}\sum_{i=1}^{\bar{n}-1}\frac{a_i}{1+a_i/c_{0,1}}+...)\}\\
&+\Phi(\beta)\prod_{j=1}^{\bar{n}-1}\frac{1}{\sqrt{1+a_j/d_{0,1}}}(1+\frac{1}{2}d_{1,1}\sum_{i=1}^{\bar{n}-1}\frac{a_i}{1+a_i/c_{0,1}}+...)\\
&\cdot\{1-\prod_{j=\bar{n}}^{n-1}\frac{1}{\sqrt{1-a_j/c_{0,2}}}(1+\frac{1}{2}c_{1,2}\sum_{i=\bar{n}}^{n-1}\frac{a_i}{1-a_i/c_{0,2}}+...)\}\end{aligned} \tag{4.3.43}$$

4.5.4 Case 4 - All a_i are positive or negative and m is odd

In order to solve the cases with an odd m and positive a_i or with an odd m and negative a_i, V is divided into two parts,

$$V=\begin{cases}V_1=\sum_{i=1}^{n-1}a_iy_i^m, & y_i>0\\ V_2=\sum_{i=1}^{n-1}a_iy_i^m, & y_i\le 0\end{cases} \tag{4.3.44}$$

Since m is odd, V_1 is positive and V_2 is negative if all a_i are positive, and V_1 is negative and V_2 is positive if all a_i are negative. Assuming that all a_i are positive, and $V_1>0$ and $V_2<0$, the function $f(V)$ is expanded as

$$\Phi(\beta+V)-\Phi(\beta) = \begin{cases} \Phi(-\beta)\{1-exp(-\frac{V_1}{c_{0,1}})(1+c_{1,1}V_1+c_{2,1}V_1^2+c_{3,1}V_1^3+...c_{N,1}V_1^N)\}, & V_1>0 \\ -\Phi(\beta)\{1-exp(\frac{V_2}{c_{0,2}})(1+c_{1,2}V_2+c_{2,2}V_2^2+c_{3,2}V_2^3+...c_{N,2}V_2^N)\}, & V_2\le 0 \end{cases} \tag{4.3.45}$$

This expansion function also satisfies Eq. (4.3.41). Based on this function, the failure probability can be computed as

$$\begin{aligned} P_f &= \frac{1}{2}+\frac{\Phi(-\beta)}{(2\pi)^{\frac{n-1}{2}}}\prod_{j=1}^{n-1}L_j^{(0)}\{1+c_{1,1}\sum_{k=1}^{n-1}\frac{L_k^{(1)}}{L_k^{(0)}}+c_{2,1}[\sum_{k=1}^{n-1}\frac{L_k^{(2)}}{L_k^{(0)}}+\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\frac{L_k^{(1)}L_l^{(1)}}{L_k^{(0)}L_l^{(0)}}] \\ &+c_{3,1}[\sum_{k=1}^{n-1}\frac{L_k^{(3)}}{L_k^{(0)}}+3\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\frac{L_k^{(2)}L_l^{(1)}}{L_k^{(0)}L_l^{(0)}}+\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\sum_{p=1,p\neq l,p\neq k}^{n-1}\frac{L_k^{(1)}L_l^{(1)}L_p^{(1)}}{L_k^{(0)}L_l^{(0)}L_p^{(0)}}]+...\} \\ &-\frac{\Phi(\beta)}{(2\pi)^{\frac{n-1}{2}}}\prod_{j=1}^{n-1}\bar{L}_j^{(0)}\{1+c_{1,2}\sum_{k=1}^{n-1}\frac{\bar{L}_k^{(1)}}{\bar{L}_k^{(0)}}+c_{2,2}[\sum_{k=1}^{n-1}\frac{\bar{L}_k^{(2)}}{\bar{L}_k^{(0)}}+\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\frac{\bar{L}_k^{(1)}\bar{L}_l^{(1)}}{\bar{L}_k^{(0)}\bar{L}_l^{(0)}}] \\ &+c_{3,2}[\sum_{k=1}^{n-1}\frac{\bar{L}_k^{(3)}}{\bar{L}_k^{(0)}}+3\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\frac{\bar{L}_k^{(2)}\bar{L}_l^{(1)}}{\bar{L}_k^{(0)}\bar{L}_l^{(0)}}+\sum_{k=1}^{n-1}\sum_{l=1,l\neq k}^{n-1}\sum_{p=1,p\neq l,p\neq k}^{n-1}\frac{\bar{L}_k^{(1)}\bar{L}_l^{(1)}\bar{L}_p^{(1)}}{\bar{L}_k^{(0)}\bar{L}_l^{(0)}\bar{L}_p^{(0)}}]+...\} \end{aligned} \tag{4.3.46}$$

where

$$\begin{aligned} L^{(0)} &= \int_0^\infty e^{-\frac{1}{2}y^2-\frac{a}{c_{0,1}}y^m}dy \\ L^{(1)} &= \int_0^\infty ay^m e^{-\frac{1}{2}y^2-\frac{a}{c_{0,1}}y^m}dy \\ L^{(2)} &= \int_0^\infty a^2y^{2m} e^{-\frac{1}{2}y^2-\frac{a}{c_{0,1}}y^m}dy \\ L^{(3)} &= \int_0^\infty a^3y^{3m} e^{-\frac{1}{2}y^2-\frac{a}{c_{0,1}}y^m}dy \end{aligned} \tag{4.3.47a}$$

$$\begin{aligned} \bar{L}^{(0)} &= \int_{-\infty}^0 e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}dy \\ \bar{L}^{(1)} &= \int_{-\infty}^0 ay^m e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}dy \\ \bar{L}^{(2)} &= \int_{-\infty}^0 a^2y^{2m} e^{-\frac{1}{2}y^2+\frac{a}{c_{0,2}}y^m}dy \end{aligned}$$

$$\bar{L}^{(3)} = \int_{-\infty}^{0} a^3 y^{3m} e^{-\frac{1}{2}y^2 + \frac{a}{c_{0,2}} y^m} dy \tag{4.3.47b}$$

Since the integrations of Eqs. (4.3.47a) and (4.3.47b) are from 0 to ∞ and from $-\infty$ to 0, a nine-term Laguerre approximate integral formula is used to compute the integral of Eq. (4.3.47).

$$L^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(1)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2\bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(3)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3\bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m} \tag{4.3.48a}$$

$$\bar{L}^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(1)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2\bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(3)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3\bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m} \tag{4.3.48b}$$

where $\bar{\lambda}_i$ and $\bar{\xi}_i$ are Laguerre integral parameters which are given in Table E2 of Appendix E.

If all a_i are negative, and $V_1 < 0$ and $V_2 > 0$, the failure probability formula is similar to Eq. (4.3.46). But the integration interval of Eq. (4.3.47a) is from $-\infty$ to 0 for $L^{(i)}(i = 0, 1, 2, 3)$ and Eq.(4.3.47b) is from 0 to ∞ for $\bar{L}^{(i)}(i = 0, 1, 2, 3)$. Eq. (39a) becomes

$$L^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(1)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2 \bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(3)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3 \bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m} \tag{4.3.49a}$$

Eq. (39b) becomes

$$\bar{L}^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(1)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2 \bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(3)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3 \bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m} \tag{4.3.49b}$$

4.5.5 Case 5 - Some a_i are positive, some a_i are negative, and m is odd

If some a_i are positive, some a_i are negative, and m is odd, V is divided as

$$V = \begin{cases} V_1 = u_1 + u_2, & y_i > 0 \\ V_2 = \bar{u}_1 + \bar{u}_2, & y_i \leq 0 \end{cases} \tag{4.3.50a}$$

where

$$u_1 = \sum_{i=1}^{\bar{n}-1} a_i y_i^m > 0, \qquad a_i > 0 \quad (i = 1, 2, ..., \bar{n}-1), \quad y_i > 0 \tag{4.3.50b}$$

$$u_2 = \sum_{i=\bar{n}}^{n-1} a_i y_i^m < 0, \qquad a_i < 0 \quad (i = \bar{n}, \bar{n}+1, ..., n-1), \quad y_i > 0 \tag{4.3.50c}$$

$$\bar{u}_1 = \sum_{i=\bar{n}}^{n-1} a_i y_i^m > 0, \qquad a_i < 0 \quad (i = \bar{n}, \bar{n}+1, ..., n-1), \quad y_i < 0 \tag{4.3.50d}$$

$$\bar{u}_2 = \sum_{i=1}^{\bar{n}-1} a_i y_i^m < 0, \qquad a_i > 0 \quad (i = 1, 2, ..., \bar{n}-1), \quad y_i < 0 \tag{4.3.50e}$$

In this equation, u_1 and $\bar{u}_1$ are positive, and u_2 and $\bar{u}_2$ are negative. By combining Case 3 and Case 4, the function $f(V)$ is expanded as

$$\Phi(\beta+V)-\Phi(\beta) = \begin{cases} \Phi(-\beta)\{1-exp(-\frac{u_1}{c_{0,1}})(1+c_{1,1}u_1+c_{2,1}u_1^2+...)\}\cdot exp(\frac{u_2}{d_{0,2}})(1+d_{1,2}u_2+...) \\ -\Phi(\beta)\{1-exp(\frac{u_2}{c_{0,2}})(1+c_{1,2}u_2+c_{2,2}u_2^2+...)\}\cdot exp(-\frac{u_1}{d_{0,1}})(1+d_{1,1}u_1+...) \\ \qquad\qquad at \quad y_i>0 \\ \\ \Phi(-\beta)\{1-exp(-\frac{\bar{u}_1}{c_{0,1}})(1+c_{1,1}\bar{u}_1+c_{2,1}\bar{u}_1^2+...)\}\cdot exp(\frac{\bar{u}_2}{d_{0,2}})(1+d_{1,2}\bar{u}_2+...) \\ -\Phi(\beta)\{1-exp(\frac{\bar{u}_2}{c_{0,2}})(1+c_{1,2}\bar{u}_2+c_{2,2}\bar{u}_2^2+...)\}\cdot exp(-\frac{\bar{u}_1}{d_{0,1}})(1+d_{1,1}\bar{u}_1+...) \\ \qquad\qquad at \quad y_i\le 0 \end{cases} \tag{4.3.52}$$

This expansion function satisfies Eq. (4.3.41). Based on this function, the failure probability can be computed as

$$\begin{aligned} P_f \;=\; & \Phi(-\beta)-\frac{\Phi(-\beta)}{(2\pi)^{\frac{n-\bar{n}}{2}}}\prod_{j=\bar{n}}^{n-1}\bar{L}_{j,d}^{(+0)}(1+d_{1,2}\sum_{k=\bar{n}}^{n-1}\frac{\bar{L}_{k,d}^{(+1)}}{\bar{L}_{k,d}^{(+0)}}+...) \\ & \cdot\{1-\prod_{j=1}^{\bar{n}-1}L_j^{(+0)}(1+c_{1,1}\sum_{k=1}^{\bar{n}-1}\frac{L_k^{(+1)}}{L_k^{(+0)}}+...)-...\} \\ & +\frac{\Phi(\beta)}{(2\pi)^{\frac{\bar{n}-1}{2}}}\prod_{j=1}^{\bar{n}-1}L_{j,d}^{(+0)}(1+d_{1,1}\sum_{k=1}^{\bar{n}-1}\frac{L_{k,d}^{(+1)}}{L_{k,d}^{(+0)}}+...)\cdot\{1-\prod_{j=\bar{n}}^{n-1}\bar{L}_j^{(+0)}(1+c_{1,2}\sum_{k=\bar{n}}^{n-1}\frac{\bar{L}_k^{(+1)}}{\bar{L}_k^{(+0)}}+...)-...\} \\ & -\frac{\Phi(-\beta)}{(2\pi)^{\frac{\bar{n}-1}{2}}}\prod_{j=1}^{\bar{n}-1}\bar{L}_{j,d}^{(-0)}(1+d_{1,2}\sum_{k=1}^{\bar{n}-1}\frac{\bar{L}_{k,d}^{(-1)}}{\bar{L}_{k,d}^{(-0)}}+...)\cdot\{1-\prod_{j=\bar{n}}^{n-1}L_j^{(-0)}(1+c_{1,1}\sum_{k=\bar{n}}^{n-1}\frac{L_k^{(-1)}}{L_k^{(-0)}}+...)-...\} \\ & +\frac{\Phi(\beta)}{(2\pi)^{\frac{n-\bar{n}}{2}}}\prod_{j=\bar{n}}^{n-1}L_{j,d}^{(-0)}(1+d_{1,1}\sum_{k=\bar{n}}^{n-1}\frac{L_{k,d}^{(-1)}}{L_{k,d}^{(-0)}}+...)\cdot\{1-\prod_{j=1}^{\bar{n}-1}\bar{L}_j^{(-0)}(1+c_{1,2}\sum_{k=1}^{\bar{n}-1}\frac{\bar{L}_k^{(-1)}}{\bar{L}_k^{(-0)}}+...)-...\} \end{aligned} \tag{4.3.53}$$

where $L_j^{(+i)}$ and $L_{j,d}^{(+i)}$ $(i=0,1,2,3, \quad j=1,2,...\bar{n}-1)$ are the integrations corresponding to u_1 $(u_1>0)$ and are given in Eqs. (4.3.47a) and (4.3.48a), but $c_{0,1}$ is replaced by $d_{0,1}$ for $L_{j,d}^{(+i)}$ integrations. $L_j^{(-i)}$ and $L_{j,d}^{(-i)}$ $(i=0,1,2,3, \quad j=\bar{n},...,n-1)$ are the integrations corresponding to $\bar{u}_1$ $(\bar{u}_1>0)$, and $L_j^{(-i)}$ can be calculated using Eq. (4.3.47a), but the integral interval is from $-\infty$ to 0. Its Laguerre formula is given in Eq. (4.3.49a). For $L_{j,d}^{(-i)}$, $c_{0,1}$ in Eq. (4.3.49a) is replaced by $d_{0,1}$. $\bar{L}_j^{(+i)}$ and $\bar{L}_{j,d}^{(+i)}$ $(i=0,1,2,3, \quad j=\bar{n},...,n-1)$ are the integrations

corresponding to u_2 ($u_2 < 0$), and $\bar{L}_j^{(+i)}$ can be computed using Eq. (4.3.47b), but the integral interval is from 0 to $+\infty$. Its Laguerre formula is given in Eq. (4.3.49b). For $\bar{L}_{j,d}^{(+i)}$, $c_{0,2}$ in Eq. (4.3.49b) is replaced by $d_{0,2}$. $\bar{L}_j^{(-i)}$ and $\bar{L}_{j,d}^{(-i)}$ ($i = 0, 1, 2, 3, \quad j = 1, ..., \bar{n} - 1$) are the integrations corresponding to $\bar{u}_2$ ($\bar{u}_2 < 0$) and are given in Eqs. (4.3.47b) and (4.3.48b), but $c_{0,2}$ is replaced by $d_{0,2}$ for $\bar{L}_{j,d}^{(-i)}$.

In Ref. [12], the coefficient $c_{i,j}$ ($i = 0, 1, 2, 3, ...,\ j = 1, 2$) was calculated by comparing the same order terms (first-order, second-order and third-order derivatives) on both sides of the expansion functions. The formulae for calculating $c_{i,1}$ and $c_{i,2}$ for Case 1 and Case 2 with one-term, two-term and three-term approximations are given in the Appendix F. For Cases 3, 4 and 5, the coefficient $c_{i,1}$ is identical to the results of Case 1, and $c_{i,2}$ is identical to the results of Case 2. The coefficients $d_{0,1}$ and $d_{0,2}$ with $d_{1,1} = d_{2,1} = ... = 0$ and $d_{1,2} = d_{2,2} = ... = 0$ are calculated as

$$d_{0,1} = 2c_{0,1} \tag{4.3.54a}$$

$$d_{0,2} = 2c_{0,2} \tag{4.3.54b}$$

4.5.6 Flow-chart

The flow-chart given in Fig. 4.7 shows the scheme of the higher-order reliability method.

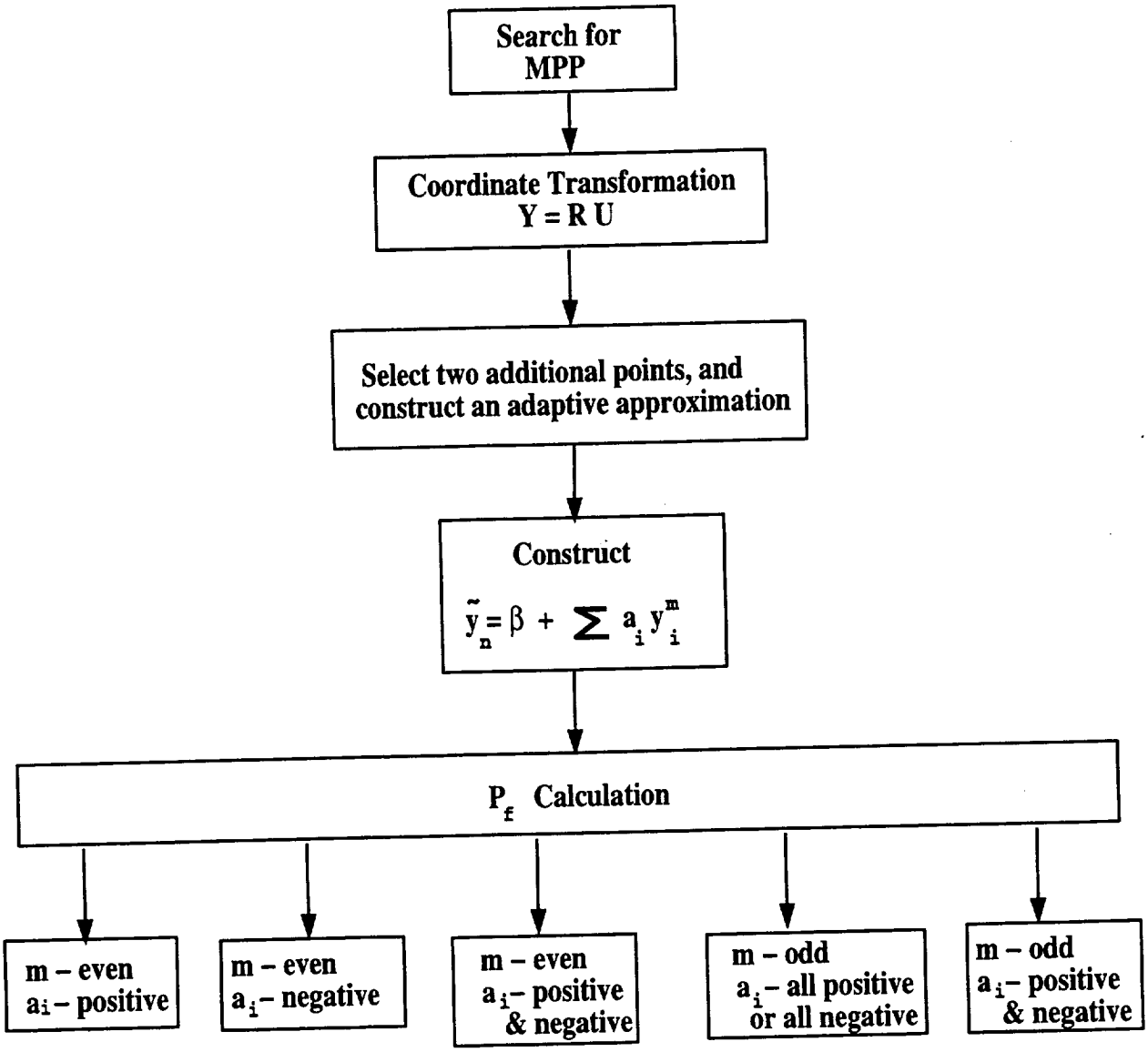


Fig. 4.7 Schematic Flow – Chart

Example 4.9

This example has a highly nonlinear performance function, which is

$$g(x_1, x_2) = 2.5 + 0.00463(x_1 + x_2 - 20)^4 - 0.2357(x_1 - x_2)$$

in which x_1 and x_2 are the random variables with normal distributions (mean $\bar{x}_1 = 10$, $\bar{x}_2 = 10$; standard deviation $\sigma_1 = \sigma_2 = 3$). Compute the failure probability using HORM.

1). Compute the safety index and the MPP, U^*;

The safety index β is calculated as $\beta = 2.5$. It requires 3 g-function and 2 gradient calculations. The MPP is located at $U^*(1.7678, -1.7678)$

2). Rotate the standard normal U-space to the new standard normal Y-space;

The orthogonal matrix can be solved by using the procedure given in Section 4.2.1.

$$H = \begin{pmatrix} \gamma_2 \\ \gamma_1 \end{pmatrix} = \begin{pmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{pmatrix}$$

Using $Y = HU$, The coordinates in the rotated standard normal space at the MPP can be obtained as $Y^*(0, 2.5)$

3). Select two additions on the limit state surface and construct the adaptive approximation.

Two points on the performance function are found as $Y_a(-0.25, 2.50588)$ and $Y_b(0.25, 2.50588)$ by using the approximate performance function which is constructed in computing the safety index. The approximation in Y-space is constructed as

$$\tilde{y}_2 = 2.5 + 1.5y_1^4$$

This approximation passes through Y_a, Y_b and Y^*, so it has a good accuracy around the MPP.

4). Calculate the failure probability using HORM;

Since the nonlinearity index m equals 4 and $a_1 = 1.5 > 0$ (Case 1), Eq. (4.3.28) is used for calculating the failure probability. The failure probability calculated by using Eq. (4.3.28) is 0.003042, which is quite close to the Monte Carlo solution ($P_f = 0.00297$). The results obtained

from Breitung and Tvedt's methods are the same as the result of FORM ($P_f = 0.006209$), which shows that there is no improvement over FORM for this highly nonlinear problem because the curvatures at the MPPs are zero.

4.6 Comparison of Different Methods

Example 4.10

The performance function in U-space is

$$g(u_1, u_2) = 1.0 + (\frac{u_1 + u_2}{a})^2 - (\frac{u_1 - u_2}{b})^2$$

in which u_1 and u_2 are the random variables with standard normal distributions; a and b can be any values. In this example, a is fixed to be 2.0. Three cases are considered with different b values (Case 1: $a = 2$, $b = \pm 0.5$; Case 2: $a = 2$, $b = \pm 2$; Case 3: $a = 2$, $b = \pm 4$)

In three cases, the MPP search is carried in U-space using an efficient safety index algorithm [6]. In this β search, the TANA2 approximation presented in Ref. [15] is used. Case 1 and Case 2 require 3 g-function and 3 gradient calculations, and Case 3 requires 8 g-function and 7 gradient calculations. The coordinates in the rotated standard normal space at the MPP are $Y^*(0, 0.3537)$, $Y^*(0, 1.4130)$, and $Y^*(0, 2.8284)$ for cases 1, 2 and 3, respectively. In order to construct the approximation in the neighborhood of the MPP, two points on the performance function are found as $Y_a(-2.1460, 0.6462)$ and $Y_b(2.1460, 0.6388)$ for case 1, $Y_a(-2.1460, 2.7024)$ and $Y_b(2.1460, 2.4460)$ for case 2, and $Y_a(-2.1460, 5.1427)$ and $Y_b(2.1460, 5.1376)$ for Case 3. This is done by using the approximate performance function which was constructed in computing the safety index. No extra exact analyses were needed. By following the procedure given in Eqs. (8-13), the approximation can be obtained as

$$\tilde{y}_2 = \begin{cases} 0.3537 + 0.06352 y_1^2, & Case1 \\ 1.4130 + 0.28000 y_1^2, & Case2 \\ 2.8284 + 0.50144 y_1^2, & Case3 \end{cases}$$

The approximate curvatures of three cases at the MPP are 0.12704, 0.5600, and 1.0029, respectively. It is obvious that the above three approximations for three cases pass through the corresponding Y_a, Y_b and Y^*, therefore they can provide good fit approximations for failure probability calculation in a larger neighborhood of the MPP. Since the nonlinearity index, m, equals 2, the closed form formula given in Eq. (4.3.31) is used to calculate the failure probability.

Fig. 4.8a depicts the iteration history of the safety index search, and Fig. 4.8b shows that the two-point adaptive approximation is closer to the exact limit state surfaces than the FORM and SORM approximations for the failure probability calculations. The numerical results of $P_f/2$ are shown in Table 4.3, which indicate that the failure probability calculated by using HORM is quite close to the Monte Carlo simulation, while the Breitung and Tvedt methods are less accurate. The Cai-Elishakoff method resulted in an impractical value for Case 3. The Koyluoglu-Nielsen and HORM gave similar P_f predictions. In HORM, there is no need for the second-order gradients information.

In addition, FORM, SORM and HORM all lose accuracy because the limit state surface is not a continuous curve in U-space or Y-space. There is no good way to predict whether or not the limit state surface will fall apart. However, one may be able to find the neglected part if some tests can be done before P_f is calculated. For example, after finding the MPP, Y_a and Y_b, we can substitute $(0, 0, ..., -\beta)$, $(-k\beta, -k\beta, ... -k\beta, -\eta_a)$, and $(k\beta, k\beta, ... k\beta, -\eta_b)$ into the limit state function. If all three points are also on the surface, it means that the limit state surface consists of two symmetrical parts. Therefore, the failure probability obtained from FORM, SORM and HORM needs to be doubled. This test requires 3 additional g-function calculations.

Example 4.11

The performance function is

$$g(x_1, x_2) = 2.2257 - \frac{0.025\sqrt{2}}{27}(x_1 + x_2 - 20)^3 + 0.2357(x_1 - x_2)$$

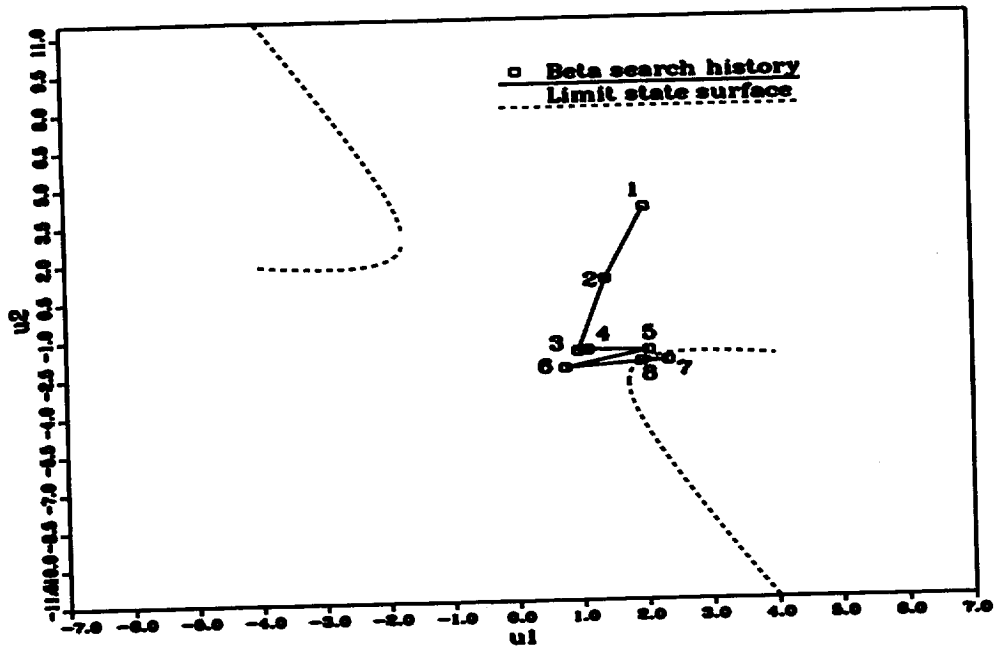


Fig. 4.8a Beta Search Using TANA2 (Case 3)

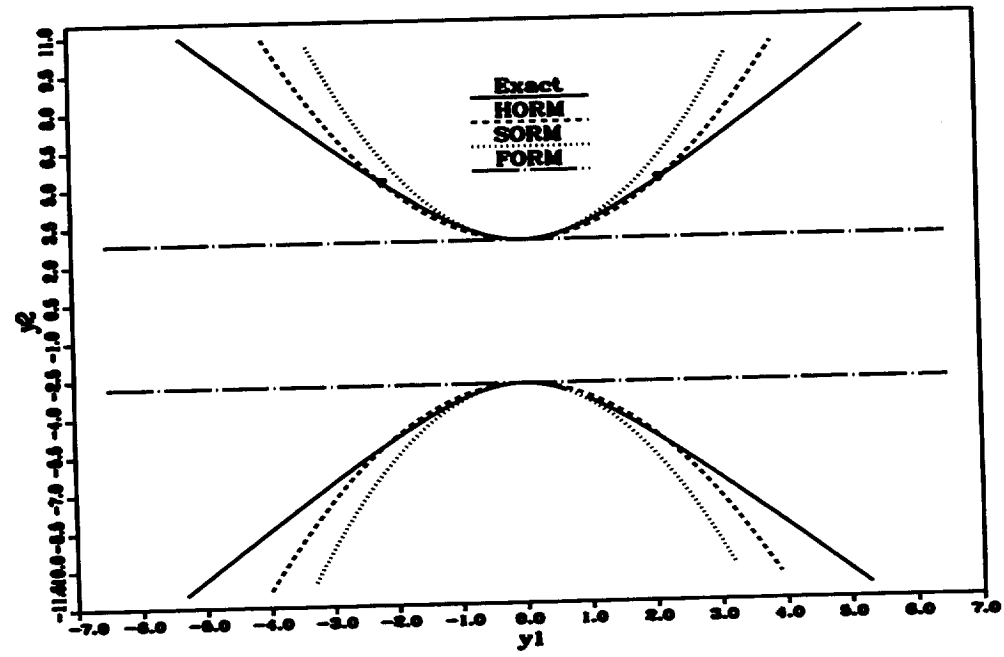


Fig. 4.8b Approximations for Pf Calculations (Case 3)

Table 4.3. Failure Probability Comparisons for Example 4.10

	Exact Analysis for P_f			Failure Probability $P_f/2$		
Method	g-func. Calculation	First-order Gradients	Second-order Gradients	Case 1	Case 2	Case 3
FORM $P_f = \Phi(-\beta)$	0	0	0	0.36179	0.07883	0.00234
Breitung	0	1	1	0.35098	0.05578	0.00105
Tvedt	0	1	1	0.34768	0.05316	0.00101
Cai-Elishakoff	0	1	1	0.33077	0.04965	-0.03269
Koyluoglu-Nielsen	0	1	1	0.33073	0.05057	0.00099
Monte Carlo	sample size=100,000			0.33659	0.05435	0.00108
HORM	0	0	0	0.33899	0.05405	0.00106

Table 4.4. Failure Probability Comparisons for Example 4.11

	Exact Analysis for P_f			
Method	g-func. Calculation	First-order Gradients	Second-order Gradients	Failure Probability P_f
FORM $P_f = \Phi(-\beta)$	0	0	0	0.013014
Breitung	0	1	1	0.013014
Tvedt	0	1	1	0.013014
Cai-Elishakoff	0	1	1	0.013014
Koyluoglu-Nielsen	0	1	1	0.013014
Monte Carlo	sample size=1,000,000			0.019188
HORM	0	0	0	0.018180

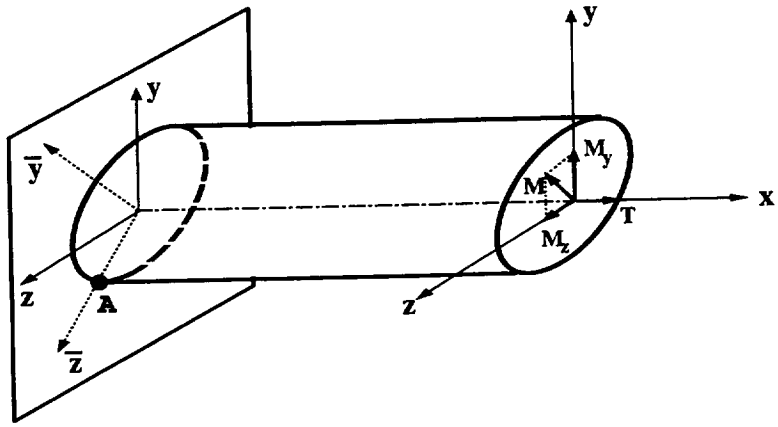


Fig. 4.9 A Circular Shaft Subjected to External Bending Moments and Torque

in which x_1 and x_2 are the random variables with normal distributions (mean $\bar{x}_1 = 10$, $\bar{x}_2 = 10$; standard deviation $\sigma_1 = \sigma_2 = 3$).

The MPP search using an efficient safety index algorithm [6] requires 3 g-function and 2 gradient calculations. The coordinates in the rotated standard normal space at the MPP are $Y^*(0, 2.2257)$. Two points on the performance function are found as $Y_a(-0.22226, 2.22471)$ and $Y_b(0.22258, 2.22691)$ by using the approximate performance function. No extra exact analyses are needed. The approximation is

$$\tilde{y}_2 = 2.2257 - 0.1y_1^3$$

Since the nonlinearity index, m, equals 3 and $a_1 = -0.1 < 0$ (Case 4), Eq. (4.3.46) is used to calculate the failure probability. The results shown in Table 4.4 indicate that the failure probability from HORM is quite close to the Monte Carlo solution ($P_f = 0.019188$), while the

results obtained from Breitung, Tvedt, and Koyluoglu-Nielsen SORM methods are the same as the FORM result. This shows these SORM methods provide no improvement over the FORM because the curvatures at the MPP are zero and the curvature-fitted paraboloid reduces to the tangent plane.

Example 4.12

This example is taken from Ref. [13], which is a circular shaft with one end clamped and one end free (Fig. 4.9). The structure is subjected to two external moments M_y and M_z and a torque T. The moments M_y, M_z and torque T are assumed as independent random variables with normal distributions. The mean values of M_y, M_z and torque T are μ_1, μ_2 and μ_3, and all the standard deviations are σ. The reliability of the shaft depends on the stresses at the point A which has the maximum tensile and shear stresses and can be calculated from

$$R = \frac{1}{(2\pi)^{3/2}} \int\int\int_{\Omega} exp[-\frac{1}{2}(u_1^2 + u_2^2 + u_3^2)] du_1 du_2 du_3$$

where u_1, u_2 and u_3 are standard normally distributed random variables of M_y, M_z and T. The safe domain Ω in X-space is described by

$$g(M_y, M_z, T) = (\frac{M_y}{\sigma})^2 + (\frac{M_z}{\sigma})^2 + (\frac{T}{\sigma})^2 - e^2 < 0$$

In U-space, the safe domain is a non-central sphere

$$g(u_1, u_2, u_3) = (u_1 + \frac{\mu_1}{\sigma})^2 + (u_2 + \frac{\mu_2}{\sigma})^2 + (u_3 + \frac{\mu_3}{\sigma})^2 - e^2 < 0$$

where $e = (\frac{M_{yield}}{\sigma})^2$, and $M_{yield}^2 \geq M_y^2 + M_z^2 + T^2$. e is the radius of the non-central sphere. An exact solution was obtained in Ref. [13] as

$$R = \Phi(r + e) - \Phi(r - e) + \frac{1}{\sqrt{2\pi r}} \{exp[-\frac{1}{2}(r + e)^2] - exp[-\frac{1}{2}(r - e)^2]\}$$

where

$$r = (\frac{\mu_1}{\sigma})^2 + (\frac{\mu_2}{\sigma})^2 + (\frac{\mu_3}{\sigma})^2$$

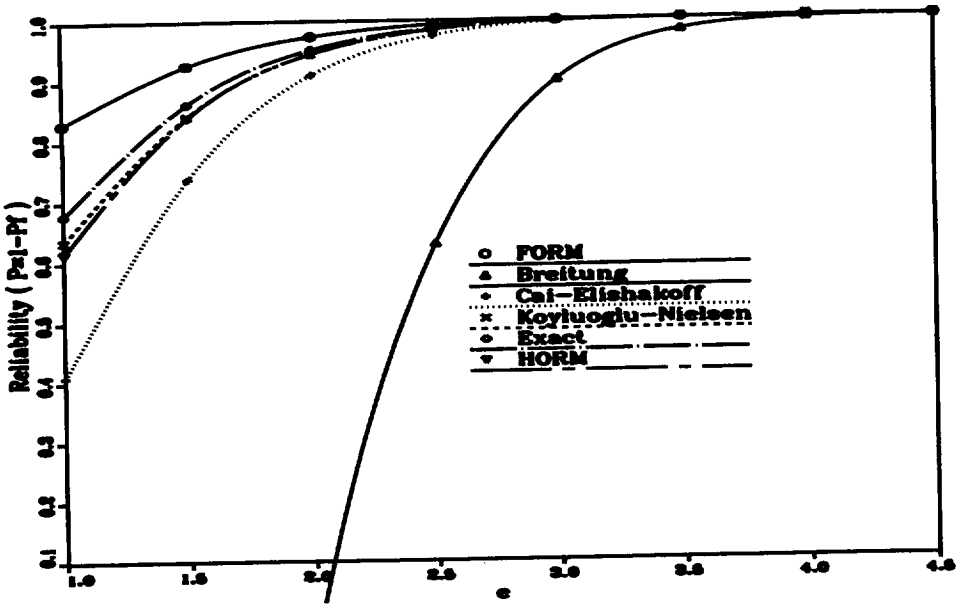


Fig. 4.10a Reliability of Example 4.12 (r=0.05)

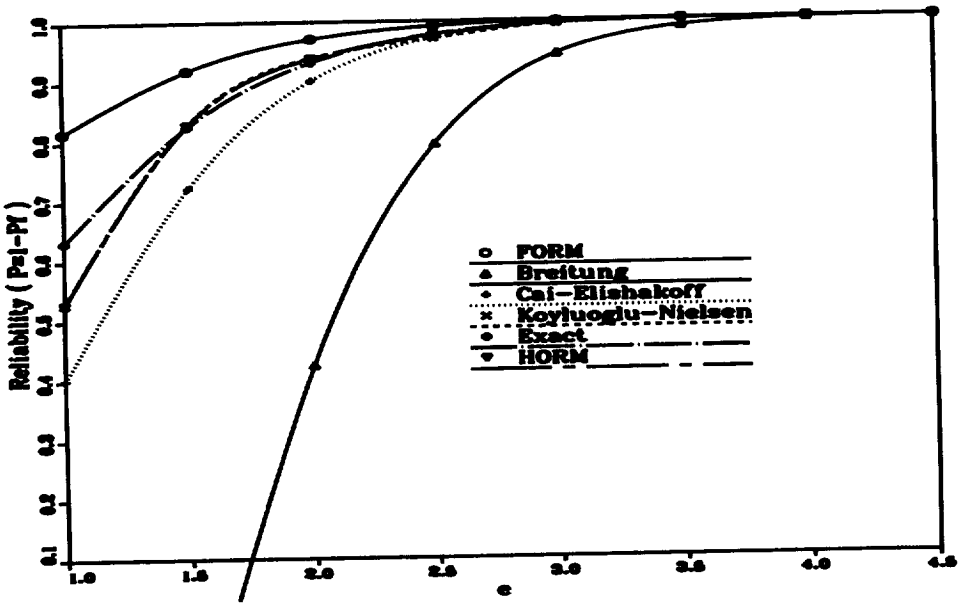


Fig. 4.10b Reliability of Example 4.12 (r=0.1)

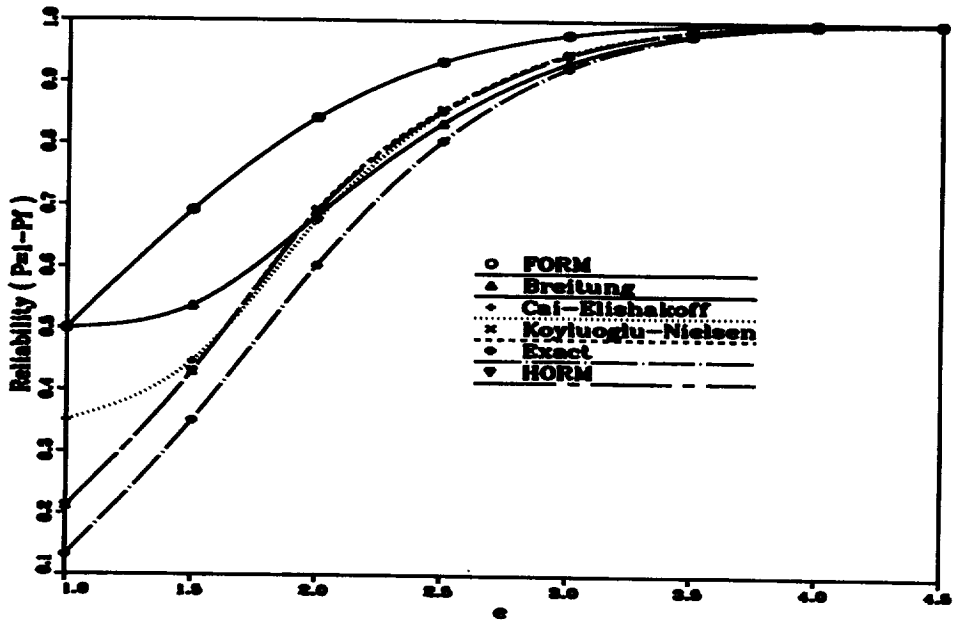


Fig. 4.10c Reliability of Example 4.12 (r=1.0)

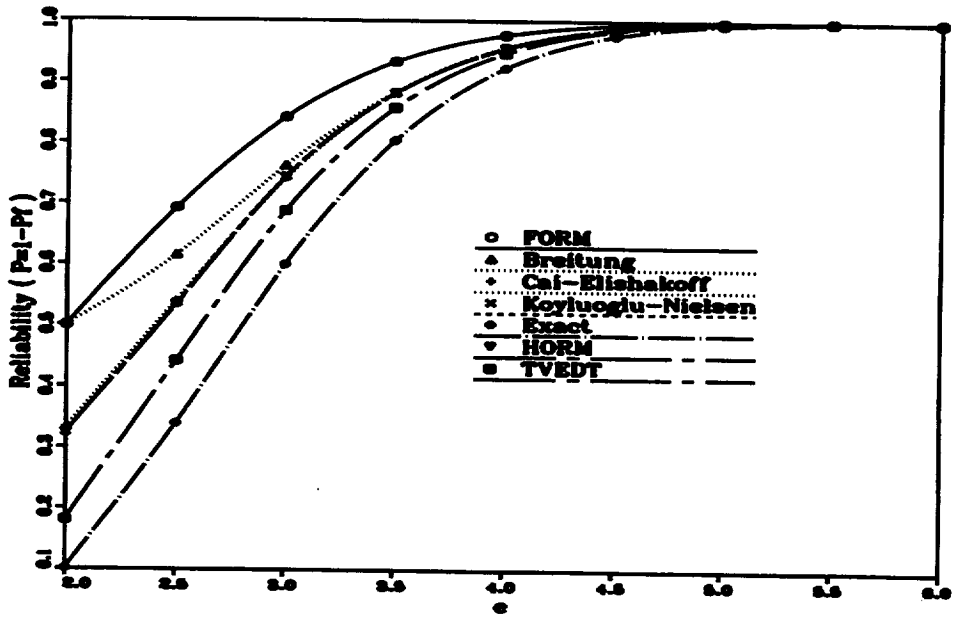


Fig. 4.10d Reliability of Example 4.12 (r=2.0)

Figs. 4.10a-4.10d show the exact and approximate reliability results of the shaft with four different values of r (Case 1: r=0.05; Case 2: r=0.1; Case 3: r=1.0; Case 4: r=2.0). For all four cases, the nonlinearity index m is equal to 2, and the closed form formula given in Eq. (4.3.37) is used for calculating the failure probability. a_i given in Eq. (4.2.27b) are calculated by the adaptive two-point approximation discussed in Section 4.2.5, so exact second-order derivatives are not required for the curvature calculations. In Figs. 4.10a-4.10c, the higher-order and Koyluoglu-Nielsen methods are the closest to the exact results even when e is small. Also, the results of HORM are almost the same as Koyluoglu-Nielsen's, which indicates that the procedure given in Eqs. (4.2.22)-(4.2.27) for constructing the approximation is very effective and accurate. The FORM gives poor accuracies when e is less than four, *i.e.* when the failure surfaces are closer to the origin. The Breitung's method yields good results when r is not small and e is greater than 2.0 for Case 3 and greater than 3.0 for Case 4, as shown in Figs. 4.10c and 4.10d. However, when r is small, the Breitung's method produces large errors; it is even worse than the FORM, as shown in Figs. 4.10a and 4.10b. Particularly, the Breitung's method resulted in impractical reliability values; for example, the probabilities are -0.0284, -1.2313 and -2.5005 when r is 2.0, 1.5 and 1.0, respectively. The Cai-Elishakoff method produces similar results to the present and Koyluoglu-Nielsen methods when e is greater than 2.5 for Cases 1 and 2, and greater than 2.0 for Cases 3 and 4. The Tvedt method is not valid for Cases 1, 2 and 3, but it provides the best results for Case 4.

4.7 Summary

In this chapter, several failure probability methods are presented. FORM can provide good results for the failure probability calculation when the limit-state surface is nearly linear in the neighborhood of the MPP. It is simple and doesn't require any additional calculations. For problems with larger curvatures, SORM needs to be used to avoid unreasonable and inaccurate FORM results. However, when the limit state function curvatures are higher at

the MPP, results obtained using SORM may be inaccurate. For example, the curvature fitted paraboloid approximations of SORM can't improve the accuracy for the cases shown in Fig. 4.4. Also, Breitung and Tvedt methods are not valid for $\beta\kappa_j \leq -1$ and do not work well in the case of negative curvatures. HORM is needed in these situations. Furthermore, HORM can be effectively used for the problems requiring expensive or impossible second-order gradient calculation of the performance function without incurring expensive computations. In HORM, the additional mathematical steps involved in building the higher-order approximations needed the Hermite and Laguerre integral formulations.

*

References

[1] Dai, Shu-Ho and Wang, Ming-O, "Reliability Analysis in Engineering Applications" Van Nostrand Reinhold, New York, 1992.

[2] Todd, J., Survey of Numerical Analysis, McGraw Hill, New York, NY, 1962.

[3] Wang, L. P. and Grandhi, R. V., "Intervening Variables and Constraint Approximations In Safety Index and Failure Probability Calculations", Structural Optimization, Vol. 10, No. 1, 1995, pp. 2-8.

[4] Der Kiureghian, A., Lin, H. Z. and Hwang, S. J., "Second Order Reliability Approximations", Journal of Engineering Mechanics, ASCE, Vol. 113, 1987, pp. 1208-1225.

[5] Grandhi, R. V. and Wang, L. P., "Higher-order Failure Probability Calculation Using Nonlinear Approximations", 37th AIAA/ ASME/ ASCE/ AHS/ ASC, Structures, Structural Dynamics, and Materials Conference, Salt Lake City, UT, AIAA 96-1461, 1996.

[6] Wang, L. P. and Grandhi, R. V., "Efficient Safety Index Calculation for Structural Reliability Analysis", Computers and Structures, Vol. 52, No. 1, 1994, pp. 103-111.

[7] Melchers, R. E., Structural Reliability Analysis and Prediction, Ellis Horwood Limited, UK., 1987.

[8] Breitung, K., "Asymptotic Approximations for Multinormal Integrals", Journal of the Engineering Mechanics Division, ASCE, Vol. 110, No. 3, Mar., 1984, pp. 357-366.

[9] Tvedt, L., "Two Second Order Approximations to the Failure Probability", Section on Structural Reliability, A/S vertas Research, Hovik, Norway, 1984.

[10] Tvedt, L., "Distribution of Quadratic Forms in Normal Space-application to Structural Reliability", Journal of the Engineering Mechanics Division, ASCE, Vol. 116, 1990, pp. 1183-1197.

[11] Hohenbichler, M. and Rackwitz, R., "Improvement of Second-order Reliability Estimates by Importance Sampling", Journal of the Engineering Mechanics Division, ASCE, Vol. 116, 1990, pp. 1183-1197.

[12] Koyluoglu, H. U. and Nielsen, S. R. K., "New Approximations for SORM Integrals", Structural Safety, Vol. 13, No. 4, April, 1994, pp. 235-246.

[13] Cai, G. Q. and Elishakoff, I., "Refined Second-order Reliability Analysis", Structural Safety, Vol. 14, No. 4, July, 1994, pp. 267-276.

[14] Bleistein, N. and Handelsman, R. A., Asymptotic Expansions of Integrals, Holt, Rinehart and Winston, New York, N. Y., 1975

[15] Wang, L. P. and Grandhi, R. V., "Improved Two-point Function Approximation for Design Optimization", AIAA Journal, Vol. 32, No. 9, 1995, pp. 1720-1727.

CHAPTER 5. SYSTEM RELIABILITY CALCULATION

The safety index and failure probability calculations in Chapters 3 and 4 were aimed primarily at assessing the reliability of one element against failing in one particular failure mode. In fact, a real structure generally consists of many elements. Even in simple structures composed of just one element, various failure modes such as bending action, shear, buckling, axial stress, deflection, etc., may exist and be relevant. The composition of many elements in structures is referred to as a "structural system". A system may be subject to many loads, either single or in various combinations. Therefore, the reliability analysis of structural systems involves consideration of multiple, perhaps correlated, limit states. Considering the structural reliability just on an individual element failure may not give a safe estimation for the structural system. A more reasonable and accurate probabilistic analysis should consider the correlation of multiple failure modes.

In this chapter, some basic definitions and concepts are introduced in Section 5.1; the failure mode approach is given in Section 5.2; the series and parallel systems are introduced in Sections 5.3 and 5.4, respectively; system reliability bounds are given in Section 5.5, and system reliability calculations using approximations are given in Section 5.6.

5.1 Basic Definitions and Concepts

5.1.1 Failure Element

In system reliability, each failure mode for a discipline or an element of a structural system is called a failure element or member. An element (or material behavior) in structural engineering is usually idealized as one of the forms of strength-deformation relationships shown in Fig. 5.1.

Elastic behavior (Fig. 5.1a) corresponds to the maximum permissible stress concept. With this idealization, failure on any one location within the structure, or of any one element, is considered to be identical to structural failure. Although this is clearly unrealistic for most

structures, it is nevertheless a convenient idealization.

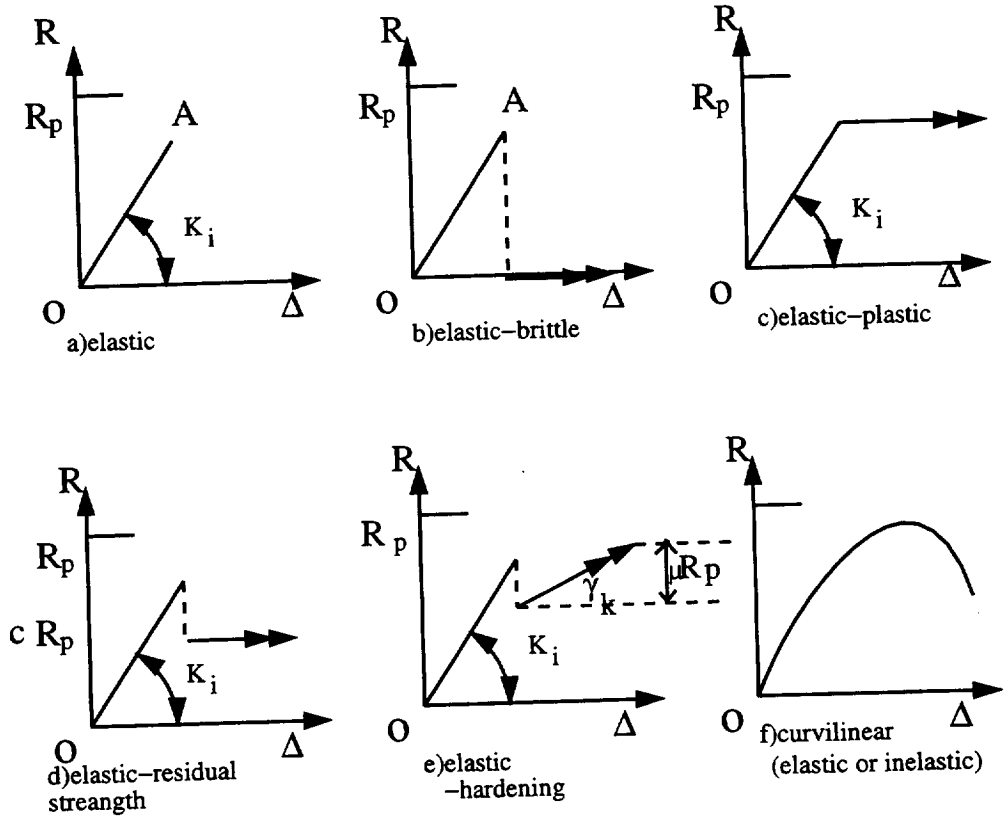


Fig. 5.1 Various strength – deformation (R–Δ) relationships

When one load system acts on a structure, the location of the peak stress or stress resultant can be identified from an elastic stress analysis. For such an analysis, use of deterministic elastic properties and dimensions is often adequate, owing to the very low coefficient of variation associated with these variables [1]. When more than one load system acts, the location of the peak stress (resultant) will depend on the relative magnitudes of the load systems, and several candidate locations or members may need to be considered. For large structures, such identification may not be easy by inspection alone.

Brittle failure of an element does not always imply structural failure, owing to redundancy of the structure. The actual member behavior can therefore be better idealized as "elastic-brittle" indicating that deformation at zero capacity is possible for a member, even after the

peak capacity has been reached (Fig. 5.1b).

Elastic-plastic element behavior (Fig. 5.1c) allows individual members or particular regions within the structure to sustain the maximum stress resultant as deformation occurs. When the elastic member's stiffness approaches infinity, this behavior is known as idealized rigid-plastic behavior. A generalization of both elastic-brittle and elastic-plastic behavior is elastic-residual strength behavior (Fig. 5.1d) and a further generalization is elastic-hardening (or softening) behavior (Fig. 5.1e). The latter may be seen as an approximation to general behavior including post-buckling effects. Even without introducing reliability concepts, the analysis of these latter behaviors is complex. Of course, general non-linear (curvilinear) strength-deformation relations (Fig. 5.1f) present even more difficulties.

5.1.2 Element Failure Probability

Element failure probability P_i^f is the failure probability of each failure element. P_i^f can be calculated using the first-order reliability method (FORM), the second-order reliability method (SORM), or the higher-order reliability method (HORM) introduced in Sections 4.3, 4.4 and 4.5, respectively.

5.1.3 Joint Failure Probability

Joint failure probability is the union probability of every two failure modes, which considers the combined effects of two or more elements. The joint failure probability can be calculated by approximating the joint failure region by the linear safety margins at the MPP of each failure surface (first-order system reliability analysis), or at the joint point U_{ij}^* on the joint failure surface closest to the origin (second-order system reliability analysis). The details of computing the joint failure probability are given in Section 5.6.

5.1.4 Structural Failure

Structural failure (as distinct from individual element or material failure) may be defined

in a number of ways, including the following:

(i) Maximum permissible stress is reached anywhere ($\sigma(x) = \sigma_{max}$);

(ii) (Plastic) collapse mechanism is formed (*i.e.* zero structural stiffness is attained);

(iii) Limiting structural stiffness is attained;

(iv) Maximum deflection is attained;

(v) Total accumulated damage reaches a limit (*e.g.* as in fatigue).

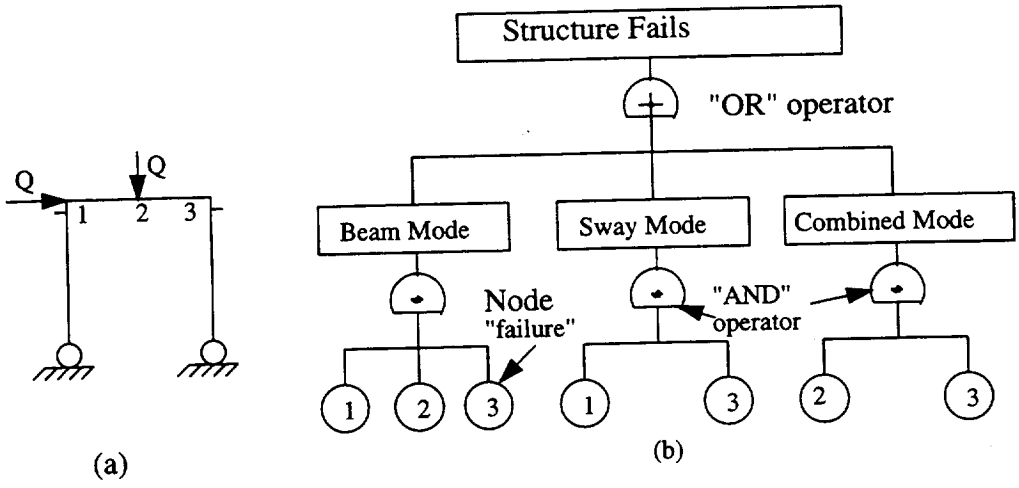


Fig 5.2 Fault–tree representation

Structural failure modes consisting of the combined effects of two or more elements or material failure events, such as statically indeterminate structures, are of particular interest in the determination of structural system reliability. If all failure modes for the system have been identified, the various events contributing to these failure modes can be systematically enumerated using the "fault-tree" concept. An example of a fault-tree is shown in Fig. 5.2b for the elementary structure of Fig. 5.2a. The procedure is to take each failure event and to decompose it into contributing subevents, which are themselves decomposed in turn. The lowest subevents in the tree correspond, for structures, to member or material failure. At this level (if not earlier) limit state equations can be written.

5.1.5 System Failure Probability

System reliability concerns the formulation of the limit states and calculation of the system failure probability when the system has more ways of failing. System failure probability includes the element failure probabilities and joint failure probabilities. The calculation of the system failure probability is given in Section 5.6.

5.1.6 Multi-dimensional Standardized Normal Distribution Function

Since the multi-dimensional standardized normal distribution function is important for computing the system failure probability, particularly the joint failure probability, the definition and mathematical details are introduced in this section. The k-dimensional standardized normal distribution function is defined as

$$\Phi_k(X;\vec{\rho}) = \int_{-\infty}^{\infty} \phi(t) \prod_{j=1}^{k} \Phi(\frac{X_j - \sqrt{\rho}t}{\sqrt{1-\rho}})dt \tag{5.1}$$

where $\vec{\rho} = [\rho_{ij}]$ is the correlation matrix for the linearized safety margins.

Assume $k = 2$ and two linear safety margins M_1 and M_2 are given as

$$M_1 = a_0 - (a_1u_1 + ... + a_nu_n) \tag{5.2a}$$

$$M_2 = b_0 - (b_1u_1 + ... + b_nu_n) \tag{5.2b}$$

where u_i $(i = 1, 2, ...n)$ are standard normal variables; M_1 and M_2 are standardized normally distributed variables with the correlation coefficient ρ, where

$$\rho = \sum_{i=1}^{n} a_ib_i \tag{5.3}$$

If $\vec{\alpha}_1 = (a_1, ..., a_n)$ and $\vec{\alpha}_2 = (b_1, ..., b_n)$ are chosen as unit vectors, the correlation coefficient ρ can be written

$$\rho = cos(v) \tag{5.4}$$

where v is the angle between the unit vectors $\vec{\alpha}_1$ and $\vec{\alpha}_2$ (Fig. 5.3, where $\rho > 0$).

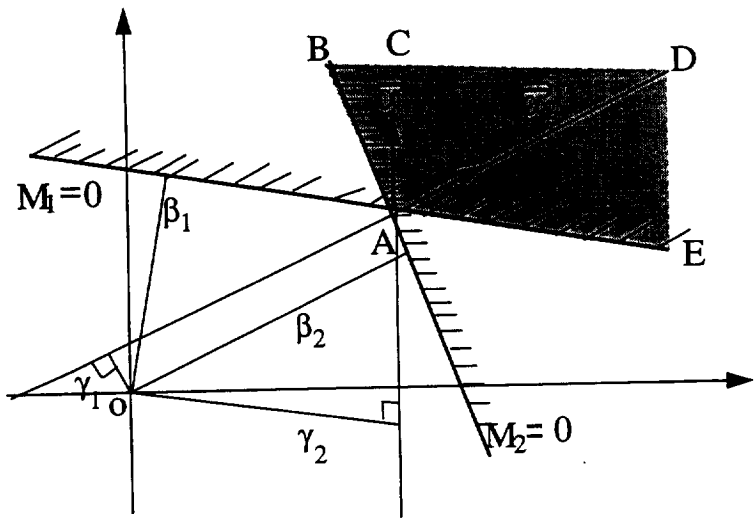


Fig 5.3 Geometrical illustration for ρ and approximate Φ_2 calculations

With the above ρ, the bivariate normal distribution function Φ_2 can be calculated as

$$\Phi_2(-\beta_i, -\beta_j; \rho) = \Phi(-\beta_i)\Phi(-\beta_j) + \int_0^{\rho} \phi_2(-\beta_i, -\beta_j; z)dz \tag{5.5}$$

in which $\phi_2(-\beta_i, -\beta_j; z)$ is the probability density function for a bivariate normal vector with zero mean values, unit variances, and a correlation coefficient z, which is given as

$$\phi_2(-\beta_i, -\beta_j; z) = \frac{1}{2\pi\sqrt{1-z^2}} exp[-\frac{\beta_i^2 + \beta_j^2 - 2z\beta_i\beta_j}{1-z^2}] \tag{5.6}$$

From Eqs. (5.5) and (5.6), Φ_2 must be evaluated numerically, but simple bounds on Φ_2 can be given to avoid any numerical integration. For practical purposes these bounds will generally be sufficient. Fig. 5.3 shows a situation with $\rho > 0$. The reliability indices β_1 and β_2 corresponding to the safety margins M_1 and M_2 are equal to a_0 and b_0, respectively. The joint failure probability P_{12}^f is equal to the probability content in the shaded region (angle BAE). Therefore, P_{12}^f is greater than the probability contents in the angles BAD and CAE. However, P_{12}^f is less than the sum of the probability contents in the angles BAD and CAE. This

observation makes it possible to find simple bounds for $P_{12}^f = \Phi_2(-\beta_1, -\beta_2; \rho)$. The probability content P_1 in the angle CAE is equal to

$$P_1 = \Phi(-\beta_1)\Phi(-\gamma_2) \tag{5.7}$$

and likewise the probability content P_2 in the angle BAD

$$P_2 = \Phi(-\beta_2)\Phi(-\gamma_1) \tag{5.8}$$

where γ_1 and γ_2 are shown in Fig. 5.3. By simple geometrical considerations

$$\gamma_2 = \frac{\beta_2 - \rho\beta_1}{\sqrt{1-\rho^2}} \tag{5.9}$$

$$\gamma_1 = \frac{\beta_1 - \rho\beta_2}{\sqrt{1-\rho^2}} \tag{5.10}$$

Therefore, for $\rho > 0$, the following bounds exist

$$max(P_1, P_2) \leq P_f \leq P_1 + P_2 \tag{5.11a}$$

$$max(P_1, P_2) \leq \Phi_2(-\beta_1, -\beta_2; \rho) \leq P_1 + P_2 \tag{5.11b}$$

For $\rho < 0$, the following bounds can be derived

$$0 \leq \Phi_2(-\beta_1, -\beta_2; \rho) \leq min(P_1, P_2) \tag{5.12}$$

5.2 Failure Mode Approach

The failure mode approach is based on the identification of all possible failure modes for the structure. A common example is the collapse mechanism technique for ideal plastic structures. Each mode of failure for the structure normally consists of a sequence of element "failures" (*i.e.* reaching an appropriate element limit state) sufficient to cause the structure as a whole

to reach a limit state such as (a)-(e) mentioned in Section 5.1.4. The possible ways in which this might occur can be represented by an "event tree" (Fig. 5.4) or as a "failure graph" (Fig. 5.5). Each branch of the failure graph represents the failure of an element of the structure, and any complete forward path through the branches starting from the "intact structure" node and concluding at the "failure" node represents a possible sequence of element failures. This information is also conveyed in the event tree.

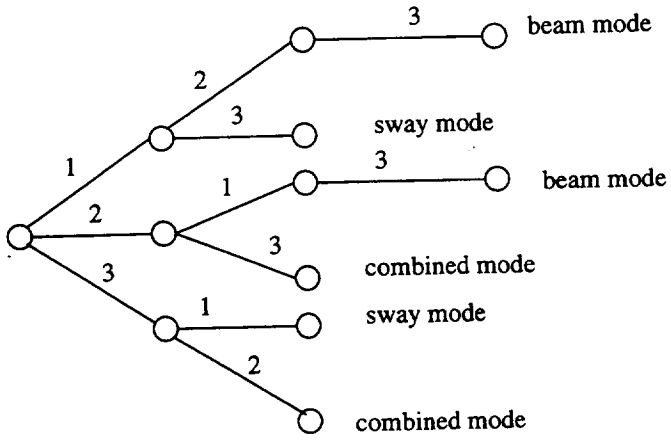


Fig. 5.4 Event – tree representation for structure of Fig. 5.2a

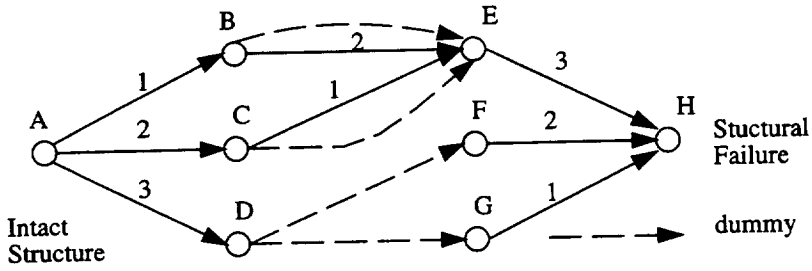


Fig. 5.5 Failure–graph representation for structure of Fig. 5.2a

Since failure through any one failure path implies failure of the structure, the event "structural failure" (F_s) is the union of all k possible failure modes:

$$P_f = P(F_s) = P(F_1 \cup F_2 \cup ... \cup F_k) \tag{5.13}$$

where F_1 is the event "failure in the ith mode". For each such mode, a sufficient number of elements (or structural "nodes") must fail; thus

$$P(F_i) = P(F_{1i} \cap F_{2i} \cap ... \cap F_{k_i'}) \tag{5.14}$$

where F_{ji} is the event "failure of the jth element in the ith failure mode" and k_i' represents the number of elements required to form the ith failure mode. For the simple example of Fig. 5.2a, there are $m = 3$ failure modes, and $k_1' = 3$, $k_2' = k_3' = 2$.

It is important to note that the failure mode approach is unconservative with respect to element failure. If the possibility of element failure (or one or more of the element failure modes) is ignored, the failure probability of the structure will usually be underestimated [2].

5.3 Series Systems

The series system is one kind of structural system idealization. In a series system, typified by a chain, and also called a "weakest link" system, attainment of any one element limit state constitutes failure of the structure (Fig. 5.6). For this idealization, the precise material properties of the elements do not matter. If the elements are brittle, failure is caused by element fracture; if the elements have a plastic deformation capacity, failure is caused by excessive yielding. It is evident that a statically determinate structure is a series system since the failure of any one of its members implies failure of the structure. Each element is therefore a possible failure mode. The system failure probability for a weakest link structure composed of k elements is computed as [3]

$$P_f = P(F_1 \cup F_2 \cup F_3 \cup ... \cup F_k) \tag{5.15}$$

Compared to Eq. (5.13), this shows that the series systems formulation given in Eq. (5.15) is of the "failure mode" type.

If each failure mode $F_i(i = 1, 2, ..., k)$ is represented by a limit state equation $g_j(X) = 0$ in X-space, the direct extension of the fundamental reliability problem given in Eq. (4.1.6a) is

$$P_f = \int_{\Omega \in X} \cdots \int f_X(X) dX \tag{5.16}$$

where X is the vector of all basic random variables and Ω is the domain in X defining failure of the system. This is defined in terms of the various failure modes as $g_j(X) \leq 0$. In two-dimensional X-space, Eq. (5.16) is defined in Fig. 5.7 with Ω and $g_j(X) \leq 0$ shown shaded.

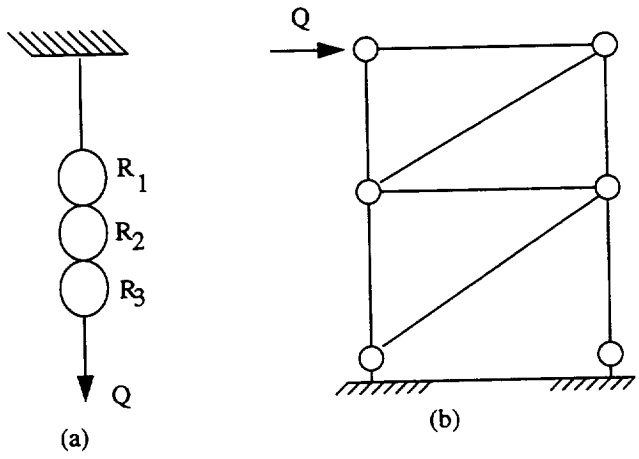


Fig. 5.6 Series system

If the limit state function $g_j(U) = 0$ in U-space corresponding to $g_j(X) = 0$ in X-space is linearized at the design point (Fig. 5.8), the failure probability of jth element can be calculated as

$$P_j^f = P(g_j(U) \leq 0) \approx P(\beta_j + \vec{\alpha}_j U \leq 0) \tag{5.17}$$

where $\vec{\alpha}_j$ is the unit normal vector at the design point and β_j is the safety index.

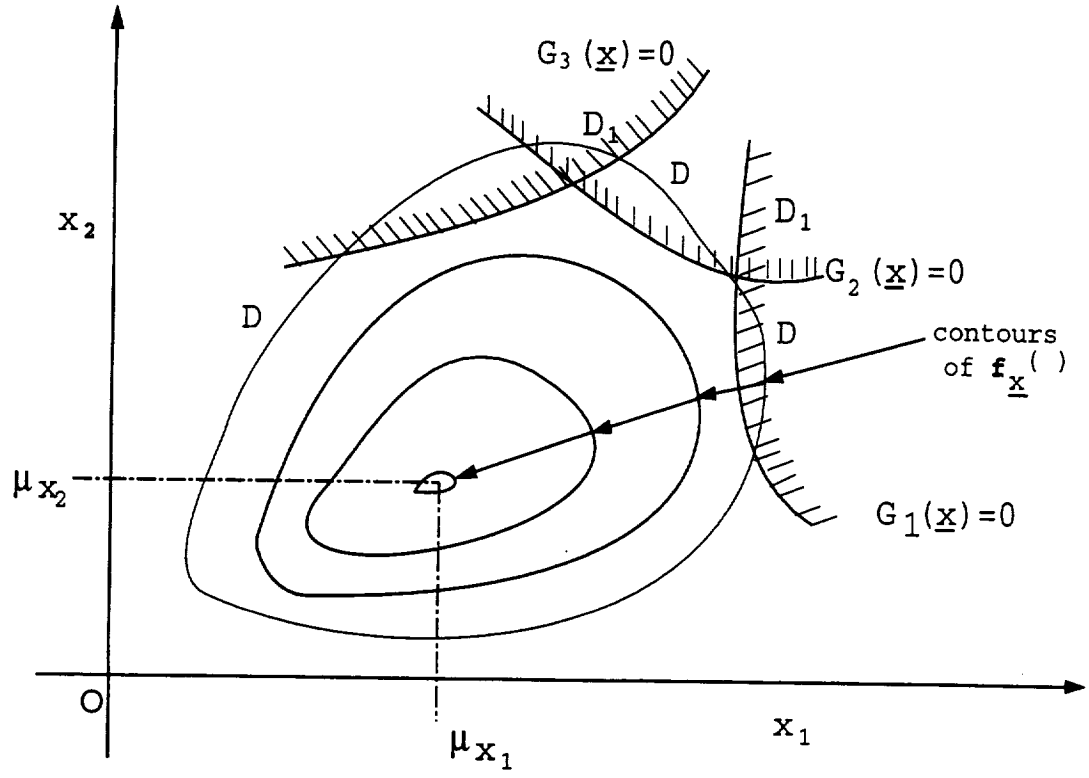


Fig. 5.7 Basic structural reliability problem in two dimensions

An approximation of the failure probability P_f of the series system shown in Fig. 5.6 can be computed as [4]

$$\begin{aligned} P_f &= P(\cup_{j=1}^{k}\{g_j(U) \le 0\}) \\ &\approx P(\cup_{j=1}^{k}\{\beta_j + \vec{\alpha}_j U \le 0\}) \\ &= P(\cup_{j=1}^{k}\{\vec{\alpha}_j U \le -\beta_j\}) \\ &= 1 - P(\cap_{j=1}^{k}\{\vec{\alpha}_j U > -\beta_j\}) \\ &= 1 - P(\cap_{j=1}^{k}\{-\vec{\alpha}_j U < \beta_j\}) \\ &= 1 - \Phi_k(\vec{\beta}; \vec{\rho}) \end{aligned} \tag{5.18}$$

where $\vec{\beta} = (\beta_1, \beta_2, ..., \beta_k)$.

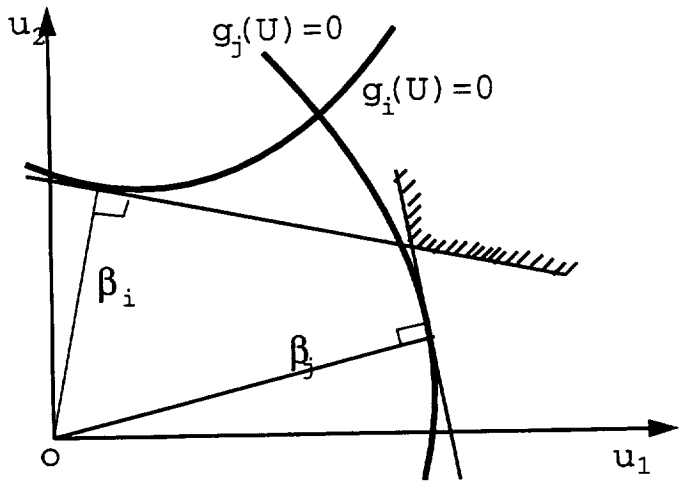


Fig. 5.8 Linearized limit state surface at the MPP in U–space

Using Eq. (5.18), the calculation of the probability of failure of a series system with linear and normally distributed safety margins is reduced to calculation of a value of Φ_k. However, calculations of values of Φ_k for $k \geq 3$ can generally only be performed in an approximate way or upper and lower bounds must be used. For $k = 2$, the upper and lower bounds are given in Eqs. (5.11) and (5.12).

Example 5.1

This example is taken from Ref. [5]. The simple structural system shown in Fig. 5.9 is loaded by a single concentrated load P. Assume that system failure is failure in compression in element 1 or element 2. Let the load-carrying capacity in the elements 1 and 2 be 1.5 n_F and assume that P and n_F are realizations of independent normally distributed random variables P and N_F with

$$\mu_P = 4kN, \qquad \sigma_P = 0.8KN$$

$$\mu_{N_F} = 4kN, \qquad \sigma_{N_F} = 0.4kN$$

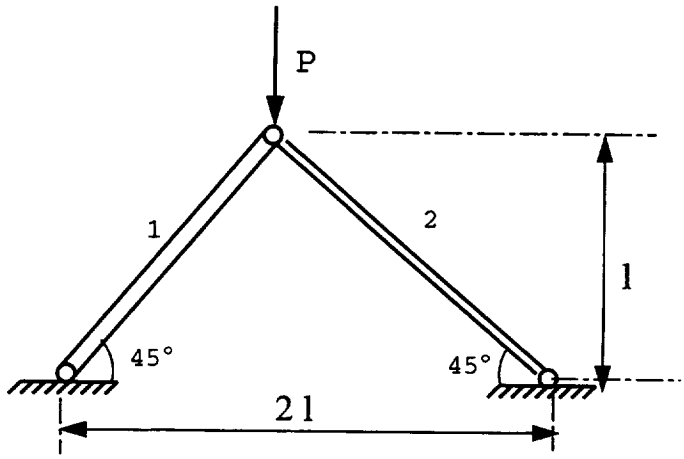


Fig. 5.9 Simple structural system with a single concentrated load

Safety margins for elements 1 and 2 are

$$M_1 = \frac{3}{2}N_F - \frac{\sqrt{2}}{2}P$$

$$M_2 = N_F - \frac{\sqrt{2}}{2}P$$

Since M_1 and M_2 are linear functions, the safety indices of M_1 and M_2 can be easily computed using the Mean Value method (Eq. (3.3.7)).

$$\begin{aligned}\beta_1 &= \frac{\mu_{M_1}}{\sigma_{M_1}} \\ &= \frac{6-2\sqrt{2}}{\sqrt{(\frac{3}{2}\times 0.4)^2 + (\frac{\sqrt{2}}{2}\times 0.8)^2}} \\ &= 3.846\end{aligned}$$

$$\begin{aligned}\beta_2 &= \frac{\mu_{M_2}}{\sigma_{M_2}} \\ &= \frac{4-2\sqrt{2}}{\sqrt{(\times 0.4)^2 + (\frac{\sqrt{2}}{2}\times 0.8)^2}} \\ &= 1.691\end{aligned}$$

$$\vec{\alpha}_1 = (0.728, -0.686)$$

$$\vec{\alpha}_2 = (0.577, -0.816)$$

So the U-space safety margins can be given as

$$M_1 = 0.728u_1 - 0.686u_2 + 3.846$$

$$M_2 = 0.577u_1 - 0.816u_2 + 1.691$$

The correlation coefficient between the safety margins is

$$\rho = 0.728 \times 0.577 + 0.686 \times 0.816 = 0.98$$

Therefore, the probability of failure of the system is

$$P_f = 1 - \Phi_2(3.846, 1.691; 0.98)$$

5.4 Parallel Systems

When the elements in a structural system (or subsystem) behave in such a way or are so interconnected that the reaching the limit state in any one or more elements does not necessarily mean failure of the whole system, the reliability problem becomes one of a "parallel" or "redundant" system analysis. Parallel systems can be modeled as in Fig. 5.10.

Redundancy in systems may be of two types; "active redundancy" occurs when the redundant elements actively participate in structural behavior even at low loading and "passive redundancy" occurs when the redundant elements do not come into play until the structure has suffered a sufficient degree of degradation or failure of its elements. Passive redundancy, or "fail-safe" design, implies the availability of a reserve capacity. It increases the reliability of a system as is easily demonstrated. However, whether active redundancy is beneficial depends on

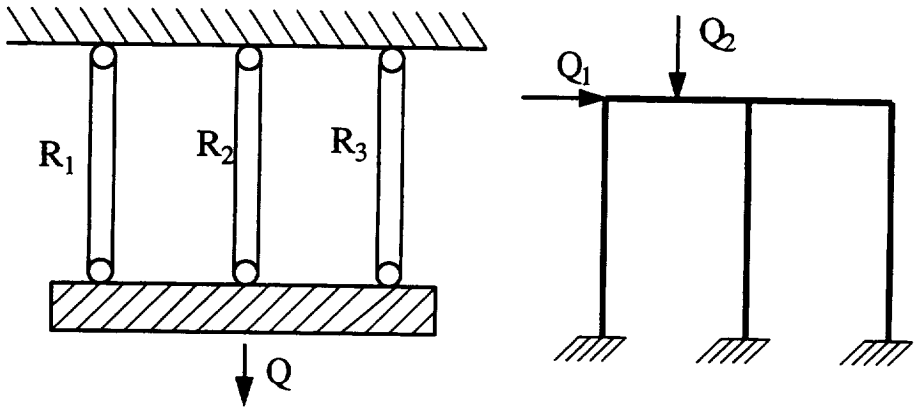


Fig. 5.10 Parallel system

the behavior characteristics of the elements. As might be expected, for ideal plastic systems, the "static theorem" guarantees that active redundancy can not reduce the reliability of a structural system.

With active redundancy, the failure probability of a k-component parallel system (or subsystem) is

$$P_f = P(F_s) = P(F_1 \cap F_2 \cap ... \cap F_k) \tag{5.19}$$

where F_i is the event "failure of the ith component". It also can be represented in X-space by

$$P_f = \int_{\Omega_1 \in X} ... \int f_X(X) dX \tag{5.20}$$

where Ω_1 is the intersection domains shown in Fig. 5.7.

Since a parallel system can only fail when all its contributory components have reached their limit states, it follows that the behavior characteristics of the components are considerably important in defining "system failure". This is in contrast with the situation for series systems.

Example 5.2

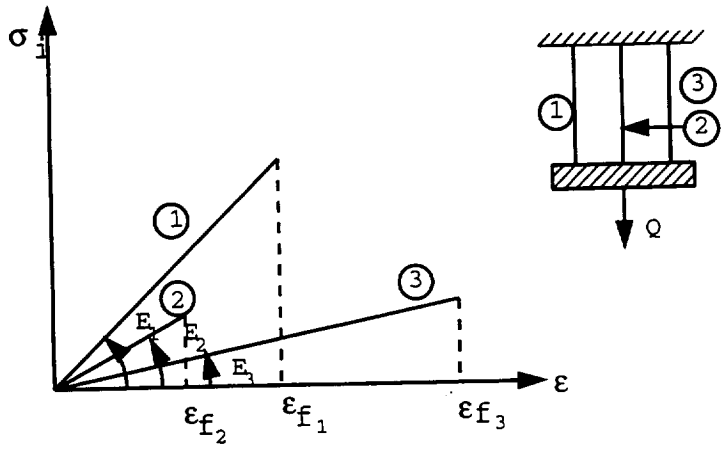


Fig. 5.11 Brittle material behavior in parallel system

This example is taken from Ref. [1] which considers the idealized parallel system shown in Fig. 5.11. If the elements are all brittle, with different fracture strains ε_f the maximum load P that can be supported at any particular strain level ε is

$$R_{sys} = \max_{\varepsilon}[R_1(\varepsilon) + R_2(\varepsilon) + R_3(\varepsilon)] \tag{5.21}$$

where $R_i = A_i\sigma_i(\varepsilon)$ for $i = 1$ to 3. Here A_i represents cross-sectional area and σ_i represents stress.

Since each resistance $R_i(i = 1,2,3)$ is a random variable, Eq. (5.21) is not easy to apply. Each possible state ε_{f1}, ε_{f2} and ε_{f3} must be considered as a possible state of maximum capacity. Thus, all possible combinations of failed and surviving elements must be considered. Each such combination will be a "parallel" subsystem, thus

$$R_{sys} = \max\{[R_1(\varepsilon_{f2}) + R_2(\varepsilon_{f2}) + R_3(\varepsilon_{f2})], [R_1(\varepsilon_{f1}) + R_3(\varepsilon_{f1})], R_3(\varepsilon_{f3})\}$$

and

$$P_f = P(R_{sys} - P < 0)$$

5.5 System Reliability Bounds

Rather than proceed with the direct integration of Eqs. (5.4) and (5.20), an alternative approach is to develop upper and lower bounds on the probability of failure of a structural system. Consider a structural system subject to a sequence of loadings and the system may fail in any one (or more) of a number of possible failure modes under any one loading in the loading sequence. The total probability of structural failure may then be expressed in terms of mode failure probabilities as

$$P_f = P(F_1) \cup P(F_2 \cap S_1) \cup P(F_3 \cap S_2 \cap S_1) \cup P(F_4 \cap S_3 \cap S_2 \cap S_1) \cup ... \tag{5.22}$$

where F_i denotes the event "failure of the structure due to failure in the ith mode, for all loading" and S_i denotes the complementary event "survival of the ith mode under all loading" (and hence survival of the structure). Since $P(F_2 \cap S_1) = P(F_2) - P(F_2 \cap F_1)...$, Eq. (5.22) can also be written as

$$P_f = P(F_1) + P(F_2) - P(F_1 \cap F_2) + P(F_3) - P(F_1 \cap F_3) - P(F_2 \cap F_3) + P(F_1 \cap F_2 \cap F_3)... \tag{5.23}$$

where $(F_1 \cap F_2)$ is the event that failure occurs in both modes 1 and 2.

5.5.1 First-order Series Bounds

The probability of failure for the structure can be expressed as $P_f = 1 - P$, where P is the probability of survival. For independent failure modes, P can be represented by the product of the mode survival probabilities, or, noting that $P_i = 1 - P_i^f$, by

$$P_f = 1 - \prod_{i=1}^{k} [1 - P_i^f] \tag{5.24}$$

where P_i^f is the probability of failure in mode i. This result can, by expansion, be shown to be identical to Eq. (5.23). It follows directly from Eq. (5.23) that, if $P_i^f \ll 1$, then Eq. (5.24)

can be approximated [3] by

$$P_f \approx \sum_{i=1}^{k} P_i^f \tag{5.25}$$

In the case where all failure modes are fully dependent, it follows directly that the weakest failure mode will always be weakest, irrespective of the random nature of the strength. Hence

$$P_f = \max_{i=1}^{k}[P_i^f] \tag{5.26}$$

Based on Eqs. (5.24) or (5.25) and (5.26), the first-order series bounds on the failure probability in terms of the individual failure mode probabilities can be given as [6]

$$\max_{i}^{k} P_i^f \leq P_f \leq \sum_{i=1}^{k} P_i^f \approx \sum_{i=1}^{k} P_i^f \tag{5.27}$$

where k is the number of failure elements, and P_i^f is the failure probability of individual failure modes.

Unfortunately for many practical structural systems, the series bounds of Eq. (5.27) are too wide to be meaningful [7]. Better bounds can be developed, but more expensive computation is required.

Example 5.3

Consider the structure given in Example 5.1, with two failure elements. The failure probability of the system is calculated as

$$P_f = 1 - \Phi_2(3.846, 1.691; 0.98)$$

where the probabilities of failure of the failure elements 1 and 2 are

$$P_1^f = \Phi(-3.846) = 0.00006$$

$$P_2^f = \Phi(-1.691) = 0.04947$$

The first-order bounds for the system failure probability are calculated as

$$0.04947 \leq P_f \leq 1 - (1 - 0.00006)(1 - 0.04947)$$

$$0.04947 \leq P_f \leq 0.04953$$

For this series system $\rho = 0.98$. Therefore, the lower bound can be expected to be close to P_f.

5.5.2 Second-order Series Bounds

Second-order bounds are obtained by retaining terms such as $P(F_1 \cap F_2)$ in Eq.(5.23), which for the case of exposition, may be rewritten as

$$\begin{aligned} P_f &= P(F_1) \\ &\quad + P(F_2) - P(F_1 \cap F_2) \\ &\quad + P(F_3) - P(F_1 \cap F_3) - P(F_2 \cap F_3) + P(F_1 \cap F_2 \cap F_3) \\ &\quad + P(F_4) - P(F_1 \cap F_4) - P(F_2 \cap F_4) - P(F_3 \cap F_4) + P(F_1 \cap F_2 \cap F_4) \\ &\qquad + P(F_1 \cap F_3 \cap F_4) + P(F_2 \cap F_3 \cap F_4) - P(F_1 \cap F_2 \cap F_3 \cap F_4) \\ &\quad P(F_5) - \ldots \\ &= \sum_{i=1}^{k} P(F_i) - \sum \sum_{i<j}^{k} P(F_i \cap F_j) + \sum \sum \sum_{i<j<l}^{k} P(F_i \cap F_j \cap F_l) - \ldots \end{aligned} \tag{5.28}$$

Because of the alternating signs as the order of the terms increases, it is evident that consideration of only first-order terms (*i.e.* $P(F_i)$) produces an upper bound on P_f, and consideration of only first- and second-order terms produces a lower bound, first-, second- and third-order terms again on upper bound, and so on.

It should also be clear that consideration of an additional failure mode can not reduce the probability of structural failure, so each complete line in Eq. (5.28) makes a non-negative

contribution to P_f. Noting that $P(F_i \cap F_j) \geq P(F_i \cap F_j \cap F_l)$,..., a lower bound to Eq. (5.28) can be obtained if only the terms $P(F_i) - P(F_i \cap F_j)$ are retained, provided each makes a non-negative contribution [8]:

$$P_f \geq P_1^f + \sum_{i=2}^{k} max\{P_i^f - \sum_{j=1}^{i-1} P(F_i \cap F_j), 0\} \tag{5.29a}$$

Let $P_{ij}^f = P(F_i \cap F_j)$, then

$$P_f \geq P_1^f + \sum_{i=2}^{k} max\{P_i^f - \sum_{j=1}^{i-1} P_{ij}^f, 0\} \tag{5.29b}$$

where $P_i^f = P(F_i)$.

An upper bound can be obtained by simplifying each line in Eq. (5.26)([8]):

$$P_f \leq \sum_{i=1}^{k} P_i^f - \sum_{i=2}^{k} \max_{j<i} P_{ij}^f \tag{5.30}$$

Therefore, the Ditlevsen second-order bounds are given as

$$P_1^f + \sum_{i=2}^{k} max\{P_i^f - \sum_{j=1}^{i-1} P_{ij}^f, 0\} \leq P_f \leq \sum_{i=1}^{k} P_i^f - \sum_{i=2}^{k} \max_{j<i} P_{ij}^f \tag{5.31}$$

The gap between the Ditlevsen bounds (Eq. 5.31) is usually much smaller than the gap between the first-order bounds given in Eq. (5.27).However, the Ditlevsen bounds require calculation of the joint probabilities P_{ij}^f and these calculations are not trivial. Usually a numerical technique must be used. When the safety margins for the failure elements i and j are linear and normally distributed, then

$$P_{ij}^f = \Phi_2(-\beta_i, -\beta_j; \rho) \tag{5.32}$$

Example 5.4

Consider a series system with two failure elements and let the reliability indices be $\beta_1 = 2.5$ and $\beta_2 = 3.0$ and let the correlation coefficient be $\rho = 0.7$. Compute the system failure

probability P_f using the first- and second-order series bounds.

1) First-order bounds

The failure probabilities of elements 1 and 2 are computed as

$$P_1^f = \Phi(-2.5) = 0.00621, \qquad P_2^f = \Phi(-3.0) = 0.00135$$

The first-order series bounds for the system failure probability are

$$0.00621 \leq P_f \leq 1 - (1 - 0.00621)(1 - 0.00135) = 0.00755$$

2) Second-order bounds

From Eqs. (5.7)-(5.12), the approximate joint failure probability P_{12}^f is calculated as

$$max(P_1, P_2) \leq \Phi_2(-\beta_1, -\beta_2; \rho) \leq P_1 + P_2$$

where

$$P_1 = \Phi(-2.5)\Phi(-\frac{3 - 2.5 \times 0.7}{\sqrt{1 - 0.7^2}}) = 3.88 \times 10^{-4}$$
$$P_2 = \Phi(-3.0)\Phi(-\frac{2.5 - 3.0 \times 0.7}{\sqrt{1 - 0.7^2}}) = 2.49 \times 10^{-4}$$

so

$$3.88 \times 10^{-4} \leq P_{12}^f \leq 6.37 \times 10^{-4}$$

If the average of the lower and upper bounds for P_{12}^f is used, *i.e.*,

$$P_{12}^f \approx 3.88 \times 10^{-4} + 6.37 \times 10^{-4} = 0.00051$$

then, the Ditlevsen bounds are

$$0.00621 + (0.00135 - 0.00051) \leq P_f \leq 0.00621 + 0.00135 - 0.00051$$

so

$$P_f = 0.00705$$

If the upper and lower bounds of P_{12}^f are used, then the Ditlevsen bounds are

$$0.00621 + 0.00135 - 0.00064 \leq P_f \leq 0.00621 + 0.00135 - 0.00039$$

i.e.

$$0.00692 \leq P_f \leq 0.00717$$

5.5.3 First-order Parallel Bounds

As in Section 5.5.1, the first-order parallel bounds are given as

$$0 \leq P_f \leq \min_i P_i^f \tag{5.33}$$

These bounds are generally of little use, and closer bounds have not been derived [9]. The computation of P_f is therefore based on Eq. (5.19) or

$$P_f = P|\cap_{i=1}^k \{g_i(U) \leq 0\}| \tag{5.34}$$

The analysis of parallel system, series system of parallel subsystems, or parallel system of series subsystems is not yet as well developed as the analysis of series systems. Some research approaches are given in Ref. [9].

5.6 System Reliability Calculation Using Approximations

In the previous section, the system reliability bounds were introduced. From Eqs. (5.27) and (5.31), it is evident that the calculations of the element failure probabilities P_i and the joint failure probabilities P_{ij} are important to obtain the system reliability bounds. As mentioned

in Chapter 4, the element failure probability P_i^f can be calculated by using first-, second- and higher-order approximations for the failure surfaces. In this section, the system reliability calculations using first-, second- and higher-order approximate failure regions are introduced.

5.6.1 Series System Reliability Calculation Using First-order Approximations

In the first-order system reliability calculation, the failure region is approximated by the polyhedral region bounded by the tangent hyperplane at the MPP. The corresponding failure probability can then be determined from Eq. (4.3.6) given in Section 4.3 for the probability contents in polyhedral regions.

5.6.1.1 First-order System Reliability Analysis Using First-order Series Bounds

If the first-order series bounds are used to compute the system probability, there is no need to calculate the joint failure probability. Thus, the first-order system failure probability using the first-order series bounds can be calculated by the following steps:

1) Compute the element failure probability P_i^f using

$$P_i^f = \Phi(-\beta_i)$$

2) Estimate system failure probability using the first-order series bounds given in Eq. (5.27);

$$\max_i P_i^f \leq P_f \leq \sum_{i=1}^{k} P_i^f$$

Example 5.5

A cantilever beam shown in Fig. 5.12 is subjected to a tip load P. Two failure modes of the displacement and stress constraints at the tip are considered as

$$g_1 = d_a - \frac{4PL^3}{Ebh^3}, \qquad displacement$$

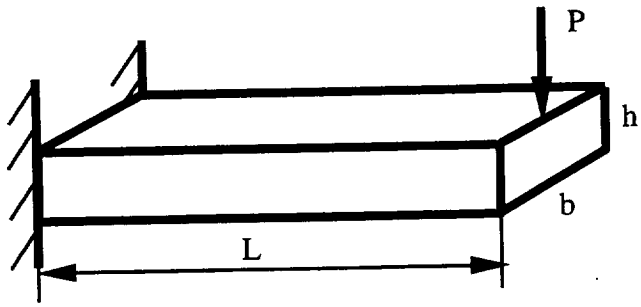


Fig. 5.12 Cantilever Beam

$$g_2 = S_a - \frac{12PL}{bh^2}, \qquad Stress$$

where d_a is the allowable displacement limit of $0.15in$; S_a is the allowable stress limit of $0.5E4psi$; L, b and h are the length, width and height of the beam, which are $30in$, $0.8359in$ and $2.5093in$, respectively. Young's modulus, E, is 10^7. The load P and the height of the beam are considered random variables with normal distributions. P has a mean of $80.0lb$, with a standard deviation of $\sigma_P = 20$; and h has a mean of 2.5093, with a standard deviation of $\sigma_h = 0.25$. The two failure modes are assumed to be dependent. Estimate the first-order system failure probability using the first-order series bounds:

1) The failure probabilities of elements 1 and 2 are calculated using $P_i^f = \Phi(-\beta_i)$:

$$P_1^f = \Phi(-\beta_1) = \Phi(-2.0922) = 0.0182$$

$$P_2^f = \Phi(-\beta_2) = \Phi(-1.9766) = 0.0241$$

2) The first-order system failure probability is estimated by the first-order series bounds:

$$\max_i P_i^f \leq P_f \leq \sum_{i=1}^{k} P_i^f$$

i.e.

$$0.0241 \leq P_f 0.0182 + 0.0241 = 0.0423$$

5.6.1.2 First-order System Reliability Analysis Using Second-order Series Bounds

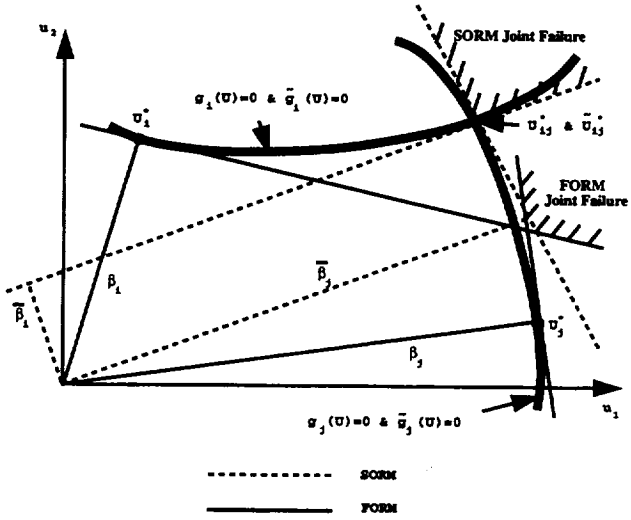


Fig. 5.13 Differences in First–order and Second–order Approximations of Joint Failure Set

If the second-order series bounds are used, there is a need to compute the first-order approximation of the joint failure probability. The first-order approximation to P_{12} is obtained by approximating the joint failure region by the linear safety margins at the design points for the two failure modes (Fig. 5.13). Fig. 5.3 shows the projection of the failure surfaces for the two failure modes on the plane spanned by the origin and the two design points u_1^* and u_2^*. Then the safety margins M_1 and M_2 are calculated using Eq. (5.2), the correlation coefficient ρ is computed using Eq. (5.3), and the joint failure probability P_{12} can be obtained by using Eq. (5.5) or the simple bounds of Eq. (5.11). The first-order system failure probability analysis using the second-order series bounds is summarized as follows

1) Compute the element failure probability P_i^f using

$$P_i^f = \Phi(-\beta_i)$$

2) Formulate the linear safety margins:

$$M_1 = \beta_1 - \sum_{k=1}^{n} a_k u_k$$

$$M_2 = \beta_2 - \sum_{k=2}^{n} b_k u_k$$

3) Compute the correlation coefficient ρ:

$$\rho = \rho[M_1, M_2] = \sum_{k=1}^{n} a_k b_k$$

4) Compute the joint failure probability P_{12} using Eq. (5.5) or Eq. (5.11).

5) Estimate the system failure probability using Eq. (5.31), that is

$$P_1^f + \sum_{i=2}^{k} max\{P_i^f - \sum_{j=1}^{i-1} P_{ij}^f, 0\} \leq P_f \leq \sum_{i=1}^{k} P_i^f - \sum_{i=2}^{k} \max_{j<i} P_{ij}^f$$

Example 5.6

Compute the first-order system failure probability for Example 5.5 using the second-order series bounds:

1) The first-order element failure probabilities are obtained as

$$P_1^f = \Phi(-\beta_1) = \Phi(-2.0922) = 0.0182$$

$$P_2^f = \Phi(-\beta_2) = \Phi(-1.9766) = 0.0241$$

2) The linear safety margins are

$$M_1 = 0.0182 + 0.4795u_1 - 0.8775u_2$$

$$M_2 = 1.9766 + 0.6294u_1 - 0.7771u_2$$

3) The correlation coefficient is obtained as

$$\rho = a_1 b_1 + a_2 b_2 = 0.4795 \times 0.6294 + (-0.8775) \times (-0.7771) = 0.9837$$

4) The joint failure probability is calculated using Eq. (5.5)

$$\begin{aligned} P_{12} &= \Phi_2(-\beta_1, -\beta_2; \rho) \\ &= \Phi(-\beta_1)\Phi(-\beta_2) + \int_0^\rho \phi_2(-\beta_1, -\beta_2; z)dz \\ &= 0.0168 \end{aligned}$$

5) The system failure probability P_f is calculated using the second-order series bounds:

$$0.0182 + 0.0241 - 0.0168 \leq P_f \leq 0.0182 + 0.0241 - 0.0168$$

so

$$P_f = 0.0255$$

5.6.2 System Reliability Calculation Using Second-order Approximations

The second-order system reliability method is based on a more accurate approximation of the failure surface than the first-order method. The element failure probabilities are computed by the second-order reliability methods (SORMs, *e.g.* Bretung [10], Tvedt [11]tvedt90,*etc.*) given in Section 4.4 of Chapter 4. The second-order approximation to the joint failure probability P_{ij} is calculated by approximating the joint failure set by the linear safety margins at the joint point U_{ij}^* on the joint failure surface closest to the origin. The difference between the first- and second-order approximations to P_{ij} is illustrated in Fig. 5.13 for a case of two basic variables. In general, u_{ij}^* will not be in the hyperplane spanned by the origin and the design points u_i^* and u_j^*; and u_{ij}^* can be found by solving the following optimization problem:

$$Min \qquad \frac{1}{2}U^T U \tag{5.35a}$$

$$s.t. \qquad g_i(U) \leq 0, \qquad g_j(U) \leq 0 \tag{5.35b}$$

Here, the constraints are formulated as inequalities, and the optimal solution does not necessarily correspond to an equality sign for both constraints. Any optimization algorithm can be used to solve this optimum problem. After solving the joint point U_{ij}^*, the safety margins can be formulated as

$$M_i = \bar{\beta}_i - \sum_{k=1}^{n} \bar{a}_k u_k \tag{5.36a}$$

$$M_j = \bar{\beta}_j - \sum_{k=1}^{n} \bar{b}_k u_k \tag{5.36b}$$

where $\bar{\beta}_i$ and $\bar{\beta}_j$ are ith and jth failure element safety indices of the linear safety margins at the joint point U_{ij}^*, and $\bar{a}_k$ and $\bar{b}_k$ are the kth direction cosines of $\bar{\beta}_i$ and $\bar{\beta}_j$, respectively. The correlation coefficient ρ is computed as

$$\rho = \sum_{k=1}^{n} \bar{a}_k \bar{b}_k \tag{5.37}$$

The joint failure probability given in Eq. (5.5) becomes

$$P_{ij}^f = \Phi(-\bar{\beta}_i)\Phi(-\bar{\beta}_j) + \int_0^{\rho_{ij}} \phi_2(-\bar{\beta}_i, -\bar{\beta}_j; z)dz \tag{5.38}$$

where

$$\phi_2(-\bar{\beta}_i, -\bar{\beta}_j; z) = \frac{1}{2\pi\sqrt{1-z^2}} exp[-\frac{\bar{\beta}_i^2 + \bar{\beta}_j^2 - 2z\bar{\beta}_i\bar{\beta}_j}{1-z^2}] \tag{5.39}$$

The main steps of computing the second-order system failure probability analysis using the second-order series bounds are given as follows:

1) Compute the element failure probability P_i^f using SORM (Section 4.4)

2) Find the joint point of the two failure surfaces U_{ij}^* by solving the optimization problem given in Eq.(5.35).

3) Formulate the linear safety margins at the joint point U_{ij}^*

$$M_1 = \bar{\beta}_1 - \sum_{k=1}^{n} \bar{a}_k u_k$$

$$M_2 = \bar{\beta}_2 - \sum_{k=2}^{n} \bar{b}_k u_k$$

4) Compute the correlation coefficient ρ

$$\rho = \rho[M_1, M_2] = \sum_{k=1}^{n} a_k b_k$$

5) Compute the joint failure probability P_{12} using Eqs. (5.38) and (5.39)

6) Estimate the system failure probability using Eq. (5.31), that is

$$P_1^f + \sum_{i=2}^{k} max\{P_i^f - \sum_{j=1}^{i-1} P_{ij}^f, 0\} \leq P_f \leq \sum_{i=1}^{k} P_i^f - \sum_{i=2}^{k} \max_{j<i} P_{ij}^f$$

Example 5.7

Compute the system failure probability for Example 5.5 using second-order approximations:

1) The second-order element failure probabilities using Tvedt's method (Eq. 4.3.17) are

$$P_1^f = 0.0173, \qquad P_2^f = 0.0232$$

2) Find the joint point of the two failure surfaces U_{ij}^* by solving the optimization problem given in Eq.(5.35)

The joint point U_{12}^* is found at $(0.6439, -2.0372)$.

3)Linearize the two failure surfaces at U_{12}^* as

$$\tilde{g}_1(U) = -0.2153(u_1 - 0.6439) + 0.375(u_2 + 2.0372)$$

$$\tilde{g}_2(U) = -0.2153(u_1 - 0.6439) + 0.25(u_2 + 2.0372)$$

Therefore, the minimum distances from the origin to $\tilde{g}_1(U)$ and $\tilde{g}_2(U)$ are

$$\bar{\beta}_1 = 2.0873, \quad \bar{\beta}_2 = 1.9638$$

The sensitivity factors of $\tilde{g}_1(U)$ and $\tilde{g}_2(U)$ are calculated as

$$\bar{a}_1 = 0.49797, \qquad \bar{a}_2 = -0.86719$$

$$\bar{b}_1 = 0.65263, \qquad \bar{b}_2 = -0.75768$$

The linear margins are

$$M_1 = 2.0873 + 0.49797u_1 - 0.86719u_2$$

$$M_2 = 1.9638 + 0.65263u_1 - 0.75768u_2$$

4) Obtain the correlation coefficient:

$$\rho = \bar{a}_1\bar{b}_1 + \bar{a}_2\bar{b}_2 = 0.49797 \times 0.65263 + (-0.86719) \times (-0.75768) = 0.982044$$

5) Compute the joint failure probability P_{12} using Eqs. (5.38) and (5.39)

$$\begin{aligned} P_{12}^f &= \Phi(-2.0873)\Phi(-1.9638) + \int_0^{0.982044} \frac{1}{2\pi\sqrt{1-z^2}} \\ &\quad exp[-\frac{2.0873^2 + 1.9638^2 - 2 \times 2.0873 \times 1.9638z}{1-z^2}] \\ &= 0.0184311 \times 0.0247787 + 0.0164805 \\ &= 0.0169372 \end{aligned}$$

6) Calculate the system failure probability P_f using the second-order series bounds:

$$0.0173 + 0.0231 - 0.0169 \leq P_f \leq 0.0173 + 0.0231 - 0.0169$$

so

$$P_f = 0.0236$$

5.6.3 System Reliability Calculation Using Higher-order Approximations

In the system reliability analysis using higher-order approximations, the element safety indices, element failure probabilities, and the joint failure probabilities are calculated using the developed two-point adaptive nonlinear approximations [16]. The joint failure set is still approximated as the linear safety margins at the joint point U_{ij}^* on the joint failure surface closest to the origin (Fig. 5.13).

The main procedure of the system reliability analysis using higher-order approximations includes:

1) Compute the safety indices of individual failure modes by approximating the failure surface using an adaptive nonlinear approximation surface at the design point [13] [14];

2) Calculate the higher-order failure probability of individual failure modes using the nonlinear approximation in a new rotated standard normal variable space, as given in Section 4.5 ([15]);

3) Find the joint point of the two failure surfaces U_{ij}^* by solving the optimization problem given in Eq.(5.35).

4) Formulate the linear safety margins at the joint point U_{ij}^*:

$$M_1 = \bar{\beta}_1 - \sum_{k=1}^{n} \bar{a}_k u_k$$

$$M_2 = \bar{\beta}_2 - \sum_{k=2}^{n} \bar{b}_k u_k$$

5) Compute the correlation coefficient ρ:

$$\rho = \rho[M_1, M_2] = \sum_{k=1}^{n} a_k b_k$$

6) Compute the joint failure probability P_{12} using Eqs. (5.38) and (5.39).

7) Estimate the system failure probability using Eq. (5.31), that is

$$P_1^f + \sum_{i=2}^{k} max\{P_i^f - \sum_{j=1}^{i-1} P_{ij}^f, 0\} \leq P_f \leq \sum_{i=1}^{k} P_i^f - \sum_{i=2}^{k} \max_{j<i} P_{ij}^f$$

The flow-chart in Fig. 5.14 shows the role of approximations in computing the safety index, element, joint and system failure probabilities. The segments utilizing the approximations are highlighted in Figure 5.14.

Example 5.8

Compute the system failure probability for Example 5.5 using higher-order approximations:

1) The higher-order element failure probabilities using HORM (Section 4.5) are

$$P_1^f = 0.0172, \qquad P_2^f = 0.0231$$

2) Find the joint point of the two failure surface U_{ij}^* by solving the optimization problem given in Eq.(5.35)

The joint point U_{12}^* is found at $(0.6439, -2.0372)$.

3)Linearize two failure surfaces at U_{12}^* as

$$\tilde{g}_1(U) = -0.2153(u_1 - 0.6439) + 0.375(u_2 + 2.0372)$$

$$\tilde{g}_2(U) = -0.2153(u_1 - 0.6439) + 0.25(u_2 + 2.0372)$$

Therefore, the minimum distances from the origin to $\tilde{g}_1(U)$ and $\tilde{g}_2(U)$ are

$$\bar{\beta}_1 = 2.0873, \quad \bar{\beta}_2 = 1.9638$$

The sensitivity factors of $\tilde{g}_1(U)$ and $\tilde{g}_2(U)$ are calculated as

$$\bar{a}_1 = 0.49797, \qquad \bar{a}_2 = -0.86719$$

$$\bar{b}_1 = 0.65263, \qquad \bar{b}_2 = -0.75768$$

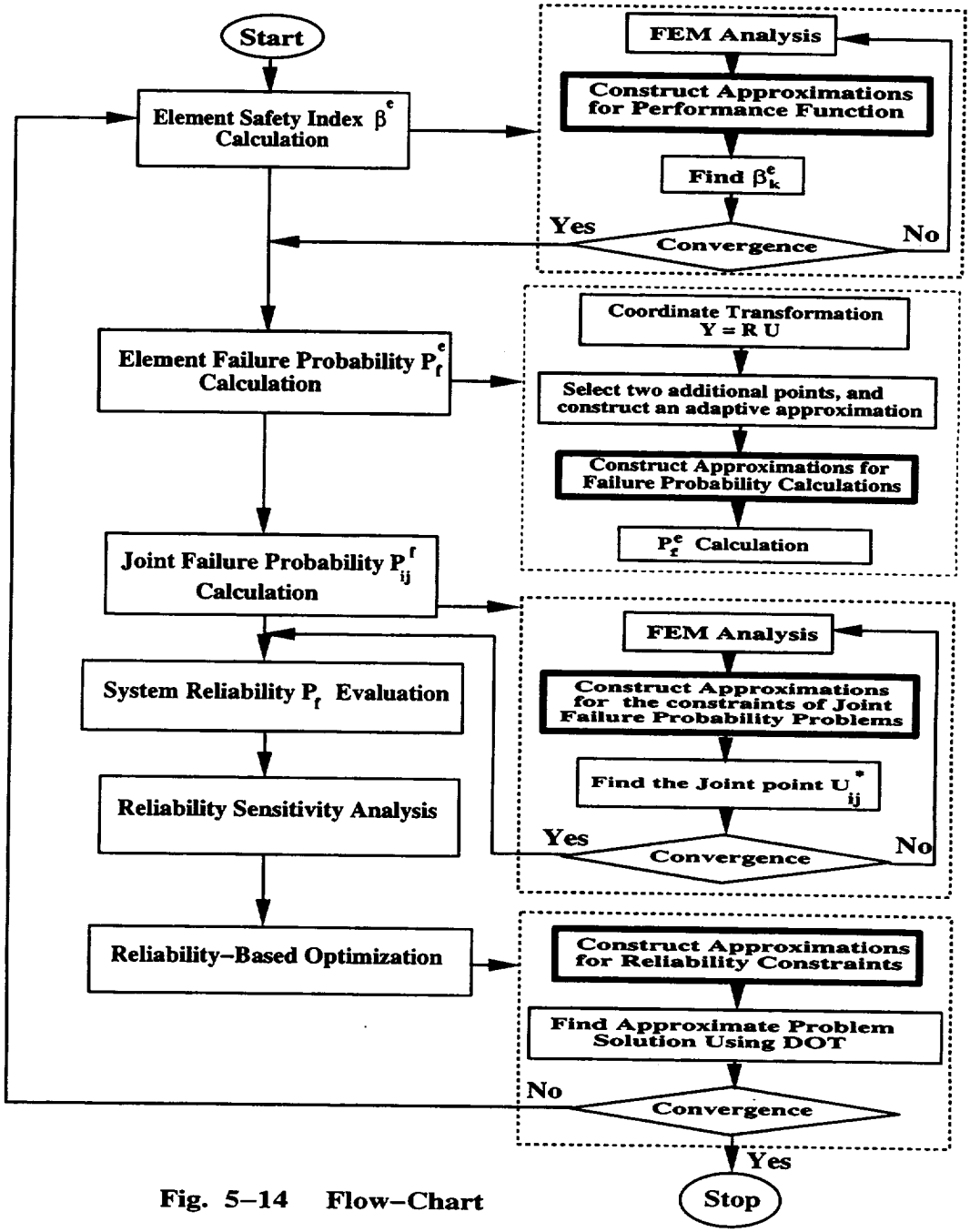


Fig. 5–14 Flow–Chart

The linear margins are

$$M_1 = 2.0873 + 0.49797u_1 - 0.86719u_2$$

$$M_2 = 1.9638 + 0.65263u_1 - 0.75768u_2$$

4) Obtain the correlation coefficient:

$$\rho = \bar{a}_1\bar{b}_1 + \bar{a}_2\bar{b}_2 = 0.49797 \times 0.65263 + (-0.86719) \times (-0.75768) = 0.982044$$

5) Compute the joint failure probability P_{12} using Eqs. (5.38) and (5.39):

$$\begin{aligned} P_{12}^f &= \Phi(-2.0873)\Phi(-1.9638) + \int_0^{0.982044} \frac{1}{2\pi\sqrt{1-z^2}} \\ &\quad exp[-\frac{2.0873^2 + 1.9638^2 - 2 \times 2.0873 \times 1.9638z}{1-z^2}] \\ &= 0.0184311 \times 0.0247787 + 0.0164805 \\ &= 0.0169372 \end{aligned}$$

6) Calculate the system failure probability P_f using the second-order series bounds:

$$0.0172 + 0.0231 - 0.0169 \leq P_f \leq 0.0172 + 0.0231 - 0.0169$$

so

$$P_f = 0.0234$$

In this example, the element failure probabilities using HORM are very close to the Monte Carlo simulation ($P_1^f = 0.0171$, $P_2^f = 0.0231$), so more accurate system failure probability is estimated. The joint failure probability P_{12}^f among FORM, SORM and HORM are quite close since the linearized surfaces at the joint point are used. More accurate calculations need to be used if more accurate results are expected. The system failure probability obtained from

FORM has some differences from SORM and HORM, even if the second-order bounds are used to calculate P_f.

The higher-order approximations used in the system failure probability calculation reduce the computational cost not only in computing the first- and second-order gradients for the element failure probabilities, but also in finding the joint point U_{12}^* for joint failure probability calculations. Without the use of approximations, finding U_{12}^* may need more than 3 iterations. As the failure elements and random variables increase, the system failure probability calculations without using approximations will be more expensive. This example shows HORM provides quite an accurate approach for the system failure probability calculations with much less computational cost.

5.7 Summary

This chapter briefly presented first- and second-order system reliability methods. The important class of series systems was analyzed by first- and second-order reliability methods using bounds on the system reliability in terms of element and joint failure mode probabilities. Because of the increased computational effort involved in calculating the joint failure probabilities, the advantages of employing high quality approximations are clear.

*

References

[1] Melchers, R. E., Structural Reliability Analysis and Prediction, Ellis Horwood Limited, UK., 1987.

[2] Bennett, R. M. and Ang. A. H.-S, Investigation of Methods for Structural Systems Reliability, Structural Research Series No. 510, University of Ilinois, Urbana, IL., 1983.

[3] Freudenthal, A. M., Garrelts, J. M., and Shinozuka, M., "The Analysis of Structural Safety", Journal of the Structural Division, ASCE, , Vol. 92, No. ST1, 1966, pp. 267-325.

[4] Hohenbichler, M. and Rackwitz, R., "First-order Concepts in System Reliability", Structural Safety, Vol. 1, 1983, pp. 177-188.

[5] Christensen P. T., and Murotsu Y., Application of Structural Systems Reliability Theory, Springer-Verlag Berlin, Heidelberg, 1986.

[6] Cornell, C. A., "Bounds on the Reliability of Structural Systems", Journal of the Structural Division, ASCE, Vol. 93, 1967, pp. 171-200

[7] Grimmelt, M. J. and Schueller, G. I., "Benchmark Study on Methods to Determine Collapse Failure Probabilities of Redundant Structures", Structural Safety, , Vol. 1, 1982, pp. 93-106.

[8] Ditlevsen, O., "Narrow Reliability Bounds for Structural Systems", Journal of Structural Mechanics, Vol. 3, 1979, pp. 453-472.

[9] Madsen, H. O., Krenk, S. and Lind, N. C., Methods of Structural Safety, Prentice-Hall, Englewood Cliffs, New Jersey, 1986.

[10] Breitung, K., "Asymptotic Approximations for Multinormal Integrals", Journal of the Engineering Mechanics Division, ASCE, Vol. 110, No. 3, Mar., 1984, pp. 357-366.

[11] Tvedt, L., "Two Second Order Approximations to the Failure Probability", Section on Structural Reliability, A/S vertas Research, Hovik, Norway, 1984.

[12] Tvedt, L., "Distribution of Quadratic Forms in Normal Space-application to Structural Reliability", Journal of the Engineering Mechanics Division, ASCE, Vol. 116, 1990, pp. 1183-1197.

[13] Wang, L. P. and Grandhi, R. V., "Efficient Safety Index Calculation for Structural Reliability Analysis", Computers and Structures, Vol. 52, No. 1, 1994, pp. 103-111.

[14] Wang, L. P. and Grandhi, R. V., "Intervening Variables and Constraint Approximations In Safety Index and Failure Probability Calculations", Structural Optimization, Vol. 10, No. 1, 1995, pp. 2-8.

[15] Grandhi, R. V. and Wang, L. P., "Higher-order Failure Probability Calculation Using Nonlinear Approximations", 37th AIAA/ ASME/ ASCE/ AHS/ ASC, Structures, Structural Dynamics, and Materials Conference, Salt Lake City, UT, AIAA 96-1461, 1996.

[16] Wang, L. P. and Grandhi, R. V., "Improved Two-point Function Approximation for Design Optimization", AIAA Journal, Vol. 32, No. 9, 1995, pp. 1720-1727.

CHAPTER 6. RELIABILITY BASED STRUCTURAL OPTIMIZATION

This chapter discusses the importance of multidisciplinary optimization and the inclusion of reliability-based constraints in design. Before addressing reliability issues, first a brief introduction of mathematical programming techniques, algorithms, sensitivity analysis, design variable linking and constraint approximations are discussed. The idea is to introduce the power of design optimization tools for minimizing the structural failure risks. Whenever the design modification for reliability improvement is involved, this in fact becomes a nested optimization. First of all, the reliability index calculation is itself an iterative process with an optimization technique for finding the shortest distance from the origin to the limit-state boundary in normal space. This optimization loop provides just the β value. At a higher level the designer would like to modify the geometries, shapes, sizes, material properties and boundary conditions to reduce the failure probability for the critical limit-states. Fig. 6.1 shows this design scheme by incorporating various steps involved in multidisciplinary optimization. Two separate iterative loops are shown, where the safety index search is an iterative process at each step of the reliability optimization.

6.1 Multidisciplinary Optimization

Multidisciplinary optimization has a maximum impact at the preliminary stage of system design. At this stage, the configuration has been defined and the materials have been selected. The design task is the determination of structural sizes that will provide an optimal structure while satisfying the numerous requirements that multiple disciplines impose on the structure. Automated structural optimization tools shorten the design cycle time and provide locally optimal designs. While, an integrated design tool brings in the requirements of diversified disciplines into a design frame work and simultaneously considers all the goals before reaching an acceptable and improved solution.

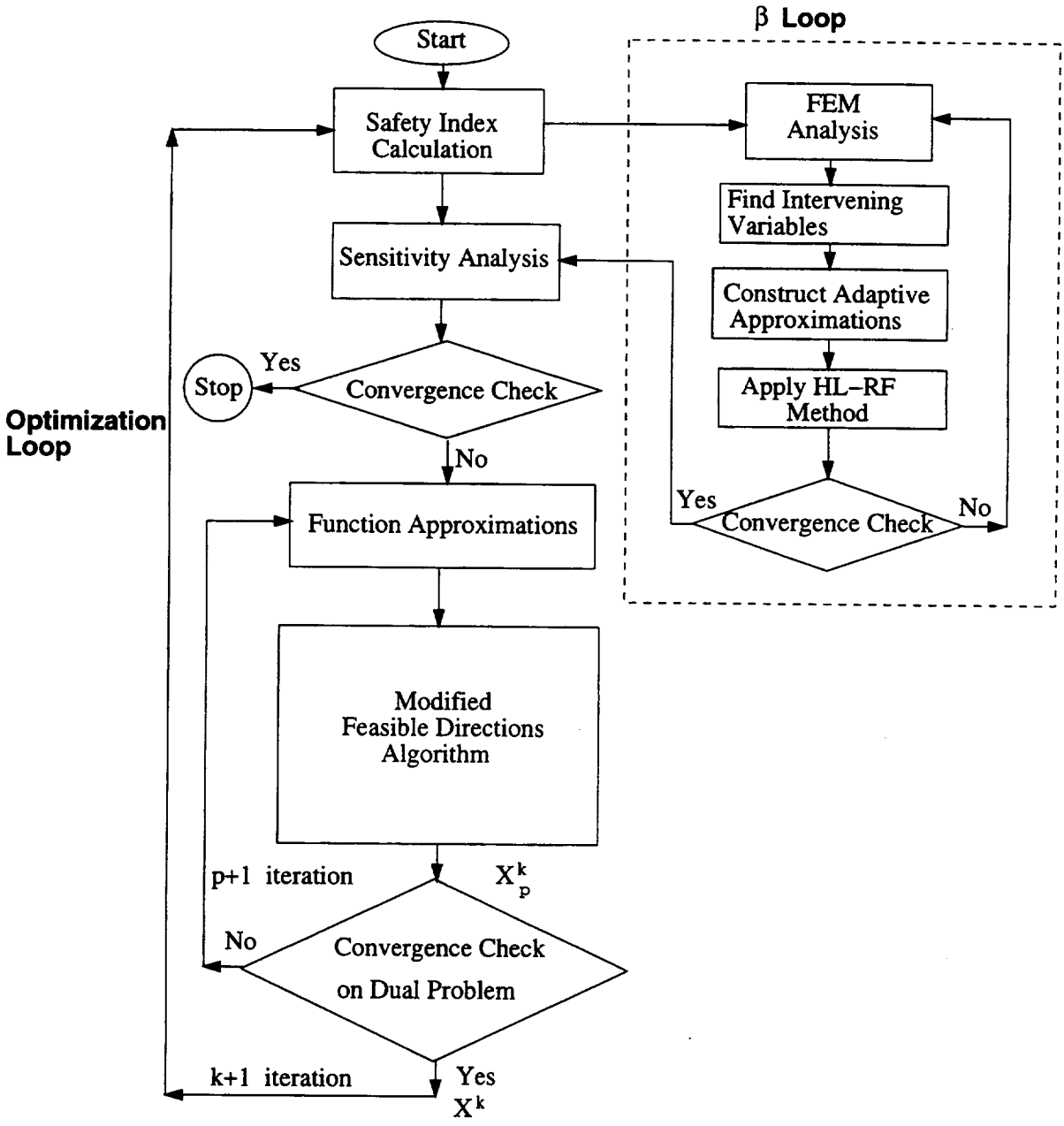


Fig.6.1 Reliability-based Optimization Algorithm Flow-Chart.

A general optimization task may be defined in a mathematical form as:

Minimize:

$$F(\vec{X}) \qquad \text{Objective} \tag{6.1}$$

Subject to:

$$g_j(\vec{X}) \leq 0 \qquad j = 1, \ldots, n_g \qquad \text{Inequality Constraints} \tag{6.2}$$

$$h_k(\vec{X}) = 0 \qquad k = 1, \ldots, n_h \qquad \text{Equality Constraints} \tag{6.3}$$

$$X_i{}^l \leq X_i \leq X_i{}^u \qquad i = 1, \ldots, n \qquad \text{Side Constraints} \tag{6.4}$$

Where

$$\vec{X} = \{X_1, X_2, \ldots, X_n\} \qquad \text{Design Variables} \tag{6.5}$$

It is with in the design space defined by the above problem statement that the optimizer searches for the best design.

There are many algorithms which can solve the above stated mathematical problem (gradient projection method, feasible directions algorithm, Lagrange multipliers method, interior and exterior penalty methods, sequential quadratic programming, etc.). Each algorithm has certain merits in solving a specific problem; however, all these methods must face the implicit nature of objective and constraint functions. For the most part, these are nonlinear functions and need computationally expensive finite element analysis. Since the solution scheme is essentially iterative, it involves a large number of structural reanalyses. Therefore the computational cost often becomes prohibitive when large scale structures are optimized with multidisciplinary requirements. To make the design problem tractable, various approximation concepts are utilized at various stages of the design steps. These include: design variable linking, temporary deletion of unimportant constraints, and the generation of high quality

explicit approximations for the implicit functions. In the following, a brief description of these efficiency improvement tools are discussed.

6.1.1 Design Variable Linking

Having an independent design variable for each free parameter or finite element gives additional degrees of freedom in solving the mathematical optimization problem. But, sometimes this results in impractical or difficult to manufacture structures. In addition, solving a problem with hundreds or thousands of design variables may not be a tractable one. Hence there are several practical advantages in reducing the number of design variables. One way is to link the local design variables with global variables. The global variables X are the ones that are directly involved in the design process. The local variables are linked to the global values through a matrix relationship of the form:

$$\vec{t} = P\vec{X} \tag{6.6}$$

where $\vec{t}$ is a vector of local variables, $\vec{X}$ is a vector of global design variables and P is the linking matrix. There are various forms of linking options possible based on the physics of the problem. The idea is to significantly reduce the number of optimization variables using the P matrix. Linking of design variables imposes additional constraints on the problem, and may not lead to the lowest possible objective function.

6.1.2 Sensitivity Analysis

Mathematical programming approaches to the solution of Equations (6.1) through (6.4) typically require the gradients of the objective function and the constraints with respect to the design variables. That is:

$$\frac{\partial F}{\partial x_i} \qquad i = 1, 2, \ldots, n \tag{6.7}$$

$$\frac{\partial g_j}{\partial x_i} \qquad j = 1, 2, \ldots, n_g, \qquad i = 1, 2, \ldots, n \tag{6.8}$$

Finite difference calculations become burdensome when there are large numbers of design variables and constraints. An efficient way to realize these gradients is to use analytical gradients. By differentiating the equations of motion for the particular analysis discipline with respect to the design variables, one could obtain the derivatives. In the case of reliability analysis, the sensitivities of the constraints are used in computing the gradients of the limit states using a chain rule of differentiation.

6.1.3 Reducing the Number of Constraints

A multidisciplinary design problem often involves a large number of inequality constraints - both behavioral and side constraints. The large number of constraints arises because it is usually necessary to guard against a wide variety of failure modes in each of several distinct loading conditions. During each stage of an iterative process, only critical and potentially critical constraints need to be considered. Non-critical and redundant constraints that are not currently influencing the iterative design process significantly are temporarily ignored. Two commonly used techniques are regionalization and "throw-away" concepts. In regionalization, for example under multiple static loading conditions, if the region contains various types of finite elements (e.g. bars, shear panels, quadrilaterals, beams) it may be desirable to retain one most critical stress constraint for each load condition and element type. The reduction of constraints by use of the regionalization concept hinges upon the assumption that the design changes made during a stage in the synthesis are not so drastic as to result in a shift of the constraint location within a region. In the "throw away" approach, unimportant (redundant or very inactive) constraints are temporarily ignored in a particular iteration.

6.1.4 Approximation Concepts

The basic objective in this approach of approximate structural analysis is to obtain high

quality algebraically explicit expressions for the objective function and behavior constraints. These explicit approximations are used in place of the detailed analysis during a stage in the iterative process. The function approximations play a very significant role in reliability-based optimization and they were extensively discussed in Chapter 3 and were successfully applied in computing the safety index, β and the failure probability, P_f. As the history shows, these concepts were extensively used in mathematical optimization and most recently were brought to the reliability analysis and design.

6.1.5 Move Limits

The approximations built at a specific point are valid with in certain bounds of n-dimensional space. In order to maintain the validity of the approximations, limits are placed on how much a local design variable can change during a design cycle. Move limits artificially restrict the design space. Proper selection of move limits is important for convergence to the optimum.

In summary the basic problem, during each stage of the iterative process, is made tractable by: (a) reducing the number of design variables through linking; (b) reducing the number of constraints via regionalization and "throw away", and (c) by constructing algebraically explicit approximations for active constraints as functions of design variables. For the purpose of demonstration, one of the widely used mathematical programming techniques, the feasible directions algorithm, is described for completeness. Using this procedure the approximate problem is solved.

6.2 Mathematical Optimization Process

The design variable vector update can be written as

$$\vec{X}^1 = \vec{X}^o + \alpha^* \vec{S}^1 \tag{6.9}$$

where $\vec{X}^o$ is the initial vector of design variables, $\vec{S}^1$ is the search vector, and α is the search

parameter. Equation (6.9) represents a one dimensional search since the update on $\vec{X}^1$ depends only on the single scalar parameter α. α^* is the value of α that yields the optimal design in the direction defined by $\vec{S}$. Finding α^* completes the first iteration in the "design" process. In order to make any further improvement in the "design,"a new search direction S^2 must be found that continues to reduce the objective function. Here we seek a "usable-feasible" direction, in which a usable direction is one that moves us downhill and a feasible direction is one that keeps us inside the bounds. This situation is shown in Fig 6.2. The mathematical definition of a usable search direction is

$$\nabla F(\vec{X}) \cdot \vec{S} \leq 0 \tag{6.10}$$

Equation (6.10) is just the scalar product (dot product)of the gradient of the objective function with the search direction. The dot product is the magnitude of $\nabla F(\vec{X})$ times the magnitude of $\vec{S}$ times the cosine of the angle between two vectors. Thus the cosine of the angle determines the sign of the product since the magnitudes are positive numbers. For the cosine to be zero or negative the angle between the vectors must be between 90 and 270 degrees. If the cosine is zero, the search direction is at an angle of 90 or 270 from the gradient of the objective function . A move in this direction follows a contour on the hill and (for a small move) does not increase or decrease the function. If the cosine is -1.0, the direction is opposite to the direction of $\nabla F(\vec{X})$ and is the direction of steepest descent. Thus we wish to find a search direction that makes the left-hand side of equation Eq. (6.10) as negative as possible. However, this direction must remain within a critical constraint. This is the feasibility requirement which is similar to the usability requirement but now is stated with respect to the constraint

$$\nabla g_j(\vec{X}) \cdot \vec{S} \leq 0 \tag{6.11}$$

Just as for the objective function, the angle between the search direction $\vec{S}$ and the gradient of the constraint must be between 90 and 270 degrees. If the angle is exactly 90 or 270 degrees, the search direction is tangent to the constraint boundary. To find the search direction, that

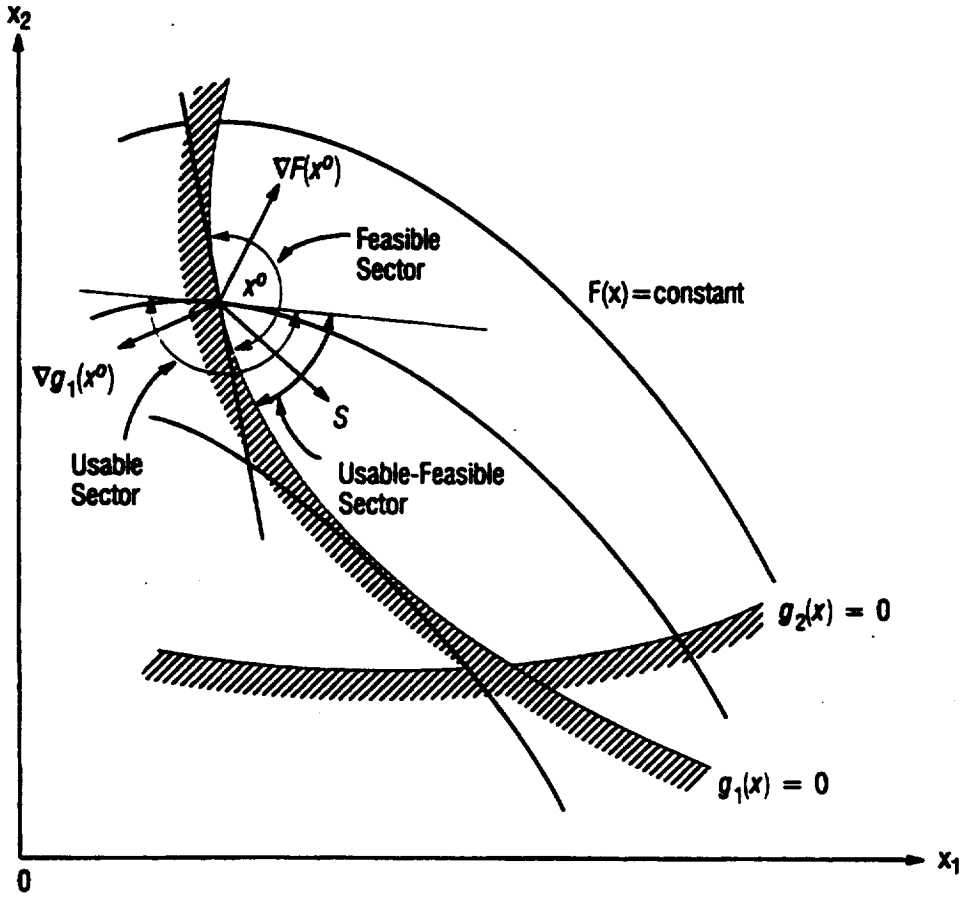


Fig.6.2 Usable-Feasible Search Direction.

makes the greatest possible improvement in the objective function but still follows or moves inside the bounds, we combine the usability and feasibility requirements. This combination creates the following sub-optimizing task: Find the components of the search direction $\vec{S}$ that minimizes

$$\nabla F(\vec{X}) \cdot (\vec{S}) \tag{6.12}$$

subject to:

$$\nabla g_j(\vec{X}) \cdot \vec{S} \leq 0 \qquad j \in J \tag{6.13}$$

$$\vec{S} \cdot \vec{S} \leq 1 \tag{6.14}$$

Where J is the set of constraints whose values are zero within some numerical tolerance. In

other words J includes any bounds that we were against, while those bounds somewhere else do not matter because we can move towards them for at least some distance without going outside. The purpose of Eq. (6.14) is simply to prevent an unbounded solution to the problem defined by Eqs. (6.12) and (6.13). In the case of a simple two-variable problem, finding the appropriate search direction is quite easy and may be done graphically. In a more general case where there are numerous design variables as several active constraints, this becomes a sub-problem that is solved as part of the optimization. This problem is linear in $\vec{S}$ except for the quadratic constraint of Eq. (6.14).

Assuming we can find a usable-feasible search direction, we can now search in this direction. If we are moving tangentially along the boundary, as is the case here, we must make corrections as we go to stay inside because the boundary is curved. That is, the constraint is assumed to be mathematically linear. We continue searching and correcting in this direction until we can make no further improvement. The sub-problem of finding a new usable-feasible search direction is repeated and is followed by continued search until no search direction can be found that improves the design without violating one or more of the constraints. We call this point the "optimum". In the present example, we began inside the bounds, and have sequentially improved the design until the optimum was reached. In practice, we start outside of one or more boundaries in which case the initial design is infeasible.

The question now arises: How do we know that we have reached the optimum? The answer can be found in what are known as the Kuhn-Tucker conditions. In the case of an unconstrained problem, this is simply the condition where the gradient of objective function vanishes (i.e., equals zero). In the case of the constrained optimization problem considered here, the conditions of optimality are more complex. Now the governing equation is the stationary condition of the Lagrangian function:

$$L(\vec{X}, \vec{\lambda}) = F(\vec{X}) + \sum_{j=1}^{M} \lambda_j g_j(\vec{X}) \tag{6.15}$$

$$\lambda_j \geq 0 \tag{6.16}$$

The Kuhn-Tucker conditions dictate that the Lagrangian function $L(X, \lambda)$ must have a vanishing gradient at the optimum design denoted by X^*. However, we must also remember the original optimization problem and the inequality conditions. When all of these conditions are considered, they lead to the statement of the Kuhn-Tucker necessary conditions for optimality:
Condition 1:

$$\vec{X}^* \text{ is feasible. Therefore, for all } j, \; G_j(\vec{X}^*) \leq 0 \tag{6.17}$$

Condition 2:

$$\lambda_j g_j(\vec{X}^*) = 0 \text{ (the product of } \lambda_j \text{ and } g_j(\vec{X}^*) \text{ equals zero)} \tag{6.18}$$

Condition 3:

$$\nabla F(\vec{X}^*) + \sum_{j=1}^{M} \lambda_j \nabla g_j(\vec{X}^*) = 0 \tag{6.19}$$

$$\lambda_j \geq 0 \qquad j = 1, 2, \ldots, M \tag{6.20}$$

The physical interpretation of these conditions is that the sum of the gradient of the objective and scalars λ_j multiplied by the associated gradients of all active constraints must vectorially add to zero. Fig. 6.3 shows this situation for a simple two-variable function space in which two constraints are active at the optimum.

These are only the necessary conditions and the definition here is actually a bit more restrictive than required in some cases. However, it does provide the essential idea. In practice, it is difficult to numerically find a design that precisely satisfies the Kuhn-Tucker conditions. Also, numerous designs may satisfy these conditions since there may be more than one constrained minimum. The importance of the Kuhn-Tucker conditions is that an understanding of the necessary conditions for optimality gives us some knowledge of what is needed to achieve an optimum. Most of the more powerful methods update the design by the relationship as shown

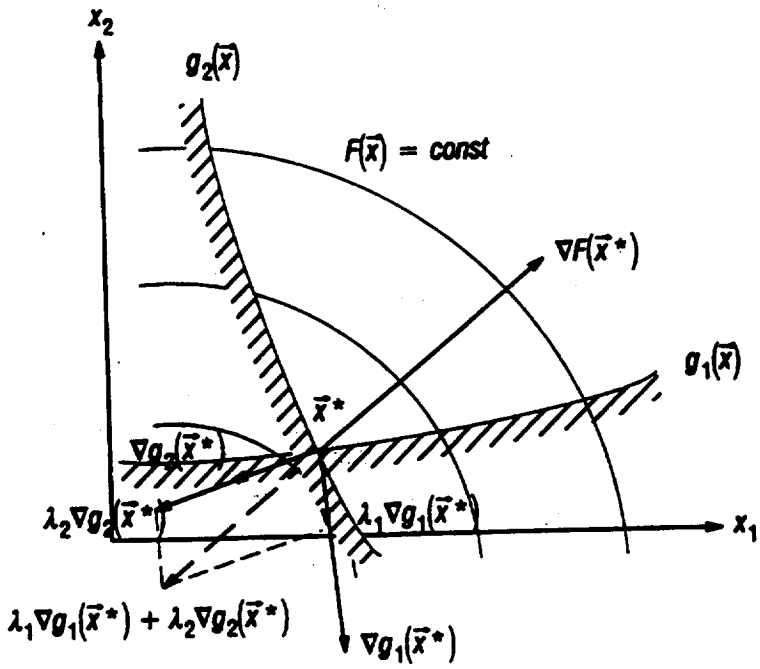


Fig.6.3 Kuhn-Tucker Conditions at a Constrained Optimum.

$$\vec{X}^q = \vec{X}^{q-1} + \alpha^* \vec{S}^q \tag{6.21}$$

where :

q = iteration number

$\vec{S}$ = vector search direction

α^* = scalar move parameter

The product $\alpha^* \vec{S}$ is the design modification at the current step. An initial design must be provided, but it does not need to satisfy all of the constraints. One of the most powerful uses of optimization is to find a feasible solution to a complicated problem.

In order to determine a search direction $\vec{S}$ which improves the design, gradients of the objective and critical constraints must be supplied. Ideally, these are computed analytically or semi-analytically. This process dramatically increases the size of the problem that can be efficiently solved. Finally, a "one-dimensional" search is performed by trying several values of

α and interpolating for the one that gives a minimum objective function while satisfying the constraints. During this process the objective function and the constraints must be repeatedly evaluated. Here the use of approximation methods plays a major role because, the evaluation of these constraints would otherwise require a full finite element analysis.

The Modified Feasible Direction Algorithm

Now we turn to the actual task of solving the approximate problem. The method described here is referred to as the Modified Method of Feasible Directions (Ref.[1]). The assumption is that we are provided with an objective function $F(\vec{X})$ and constraints $g_j(X) \leq 0, \quad j = 1, 2, \ldots, n_g$ as well as lower and upper bounds on the design variables. Also, the gradients of the objective and constraints are available. We are solving the following general problem.

Find the set of design variables $x_i, \quad i = 1, 2 \ldots, n$ contained in a vector X that minimizes

$$F(\vec{X}) \tag{6.22}$$

subject to

$$g_j(\vec{X}) \leq 0 \qquad j = 1, 2, \ldots, n_g \tag{6.23}$$

$$X_i^L \leq X_i \leq X_i^U \qquad i = 1, 2, \ldots, n \tag{6.24}$$

Given an initial X-vector X^O, update the design according to Eq. (6.21),which is also repeated here

$$\vec{X}^q = \vec{X}^{q-1} + \alpha^* \vec{S}^q \tag{6.25}$$

The optimization process now proceeds in the following steps:

1. Start, $q = 0$, $X^q = X^\circ$.
2. $q = q + 1$.
3. Evaluate $F(\vec{X})$ and $g_j(\vec{X})$ where $J = 1, 2, \ldots, n_g$.
4. Identify the set of critical and near critical constraints J.
5. Calculate $\nabla F(\vec{X})$ and $\nabla g_j(\vec{X})$ where $j \in J$.

6. Determine a usable-feasible search direction $\vec{S}^q$.

7. Perform a one-dimensional search to find α^*.

8. Set $\vec{X}^q = \vec{X}^{q-1} + \alpha^* \vec{S}^q$.

9. Check convergence to the optimum. If satisfied, exit. Otherwise, go to Step 2.

The critical parts of the optimization task consist the following:

1. Find a usable-feasible search direction, $\vec{S}^q$.

2. Find the scalar parameter α^* that minimizes $F(\vec{X})$ subjected to the constraints.

3. Test for convergence to $\vec{X}^*$ the optimum, and terminate if convergence is achieved.

We will discuss each of these in turn.

Finding the Search Direction $\vec{S}^q$

The first step in finding the search direction is to determine which constraints, if any, are active or violated. Here an active constraint is defined as one with a value between a small negative number and a small positive number. Then, the gradients of the objective function and all the active and violated constraints are calculated. Thereafter, a usable feasible search direction is found (if one exists). In this case there are three possibilities:

1. There are no active, or violated constraints.
2. There are active constraints but no violated constraints.
3. There are one or more violated constraints.

Each of these possibilities is handled differently.

No Active or Violated Constraints (Unconstrained Minimization)

Frequently at the beginning of the optimization process there are no active or violated constraints. In this case the feasibility requirement is automatically met since we can move in any direction for at least a short distance without violating any constraints. Thus, we only need to find a usable direction which is the one that points down hill. It does not have to be the steepest descent direction, but to start the process this is the preferred choice. Therefore the initial search direction is simply:

$$\vec{S}^q = -\nabla F(\vec{X}^{q-1}) \tag{6.26}$$

The steepest descent direction is only used if this is the beginning of the optimization ($q = 1$) or if the last iteration had no active or violated constraints.

Now assume the last search direction is in the steepest descent direction and there are still no active constraints. (If there were no violated constraints before, there will not be any now.) The next step can be to search again in the steepest descent direction, which is often done (This direction is perpendicular to the previous direction). We can use a "conjugate" search direction, or more precisely an A-conjugate direction where A is a matrix of second partial derivatives of the objective function. The A matrix is not actually computed but there are methods for approximating A that offer a guaranteed convergence rate for problems where the objective function is a quadratic function.

The Fletcher - Reeves conjugate direction method which is very simple and reliable is a good choice. This method defines a search direction as:

$$\vec{S}^q = -\nabla F(\vec{X}^q) + \alpha \vec{S}^{q-1} \tag{6.27}$$

$$\text{where} \qquad \alpha = \frac{| \nabla F(\vec{X}^{q-1}) |^2}{| \nabla F(\vec{X}^{q-2}) |^2} \tag{6.28}$$

The advantage of using the conjugate search direction is seen from Figures 6.4 and 6.5 which show a simple two-variable design space. In Figure 6.4 the search directions are the steepest descent directions, whereas in Figure 6.5 the conjugate directions are used. In Figure 6.4 the search direction is always perpendicular to the previous direction. On the other hand in Figure 6.5 each search direction uses the steepest descent direction plus some fraction of the previous direction.

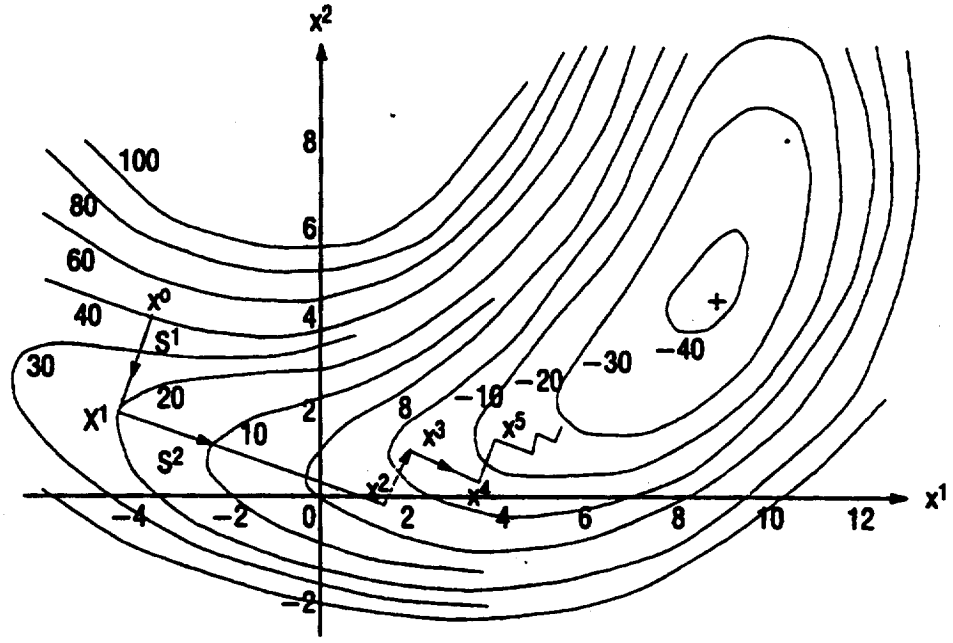


Fig.6.4 Geometric Interpretation of the Steepest Descent Method.

Active but no violated constraints

The more common search direction problem, in which the initial design is feasible and there is a constraint, is to find a search direction that improves the design but moves parallel to or away from a constraint. To solve this problem first find a search direction S that reduces the objective function without violating any currently active constraints. The following equations state the problem in mathematical terms:

Find the search direction S^q that minimizes

$$\nabla F(\vec{X}^{q-1}) \cdot \vec{S}^q \tag{6.29}$$

subject to

$$\nabla g_j(\vec{X}^{q-1}) \cdot \vec{S}^q \qquad j \in J \qquad \text{Feasibility condition} \tag{6.30}$$

$$\vec{S}^q \cdot \vec{S}^q \leq 1 \qquad \text{Bounds on } S \tag{6.31}$$

This is the same problem as shown in Eqs.(6.12) through (6.14) for finding a usable-feasible

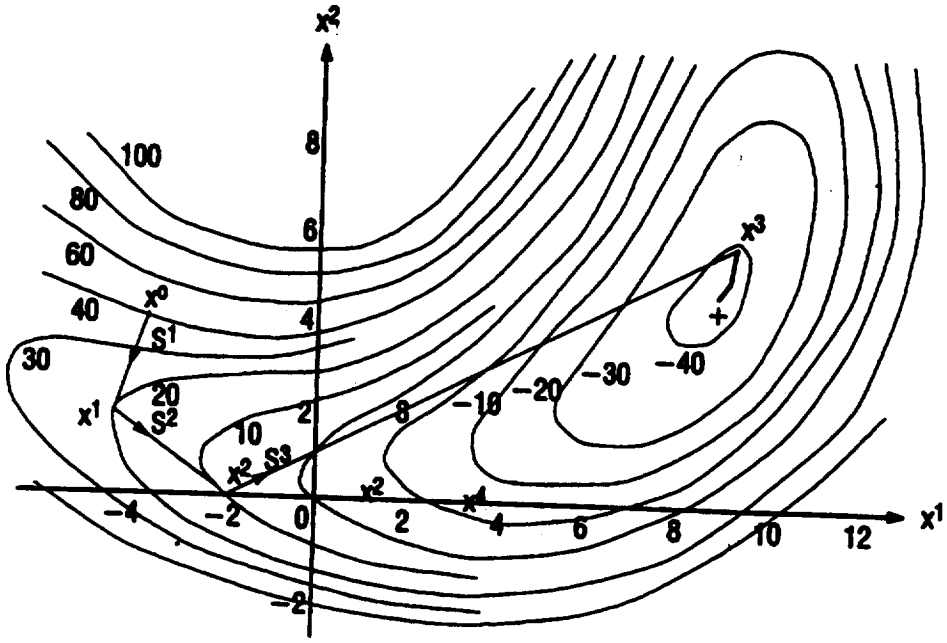


Fig.6.5 Geometric Interpretation of the Fletcher-Reeves Method.

search direction in the physical example of searching for the lowest point on the hill. Note that the scalar product is the magnitude of the two vectors multiplied by the cosine of the angle between them. Thus the objective of this sub-optimization problem is

$$| \nabla F(\vec{X}^{q-1}) | \cdot | \vec{S} | \cdot cos(\theta) \tag{6.32}$$

Since we are minimizing this function we want the cosine of the angle between the two vectors to be as large a negative number as possible but within the restriction of Eq. (6.30). Alternatively for any angle θ between 90 and 270 degrees Eq. (6.32) can be made more negative if the magnitude of $\vec{S}$ is increased. Also if $\vec{S}$ satisfies the requirements of Eq. (6.30), then any increase in magnitude of $\vec{S}$ also satisfies this equation. Therefore it is necessary to bound $\vec{S}$ which is accomplished using Eq. (6.31).

Assuming the resulting objective function from this subproblem is negative, the usability requirement of Eq(6.10) is satisfied. If the objective function defined by Eq. (6.29) cannot be forced to be negative, then it follows that no direction exist that reduces the objective function

while staying inside the constraints. If this is the case the Kuhn-Tucker conditions are satisfied, and the optimization process may be terminated.

One or more violated constraints

Now consider the case where one or more constraints are violated. Such a case is shown in Figure 6.6 where constraint $g_1(X)$ is violated and $g_2(X)$ is active. Now we must find a search direction back toward the feasible region even if it is necessary to increase the objective function to do so. To achieve this, we augment over the direction-finding problem of Eqs.(6.29) through (6.31) with a new variable W. This process has no direct physical significance to the problem except as a measure of the constraint violation.

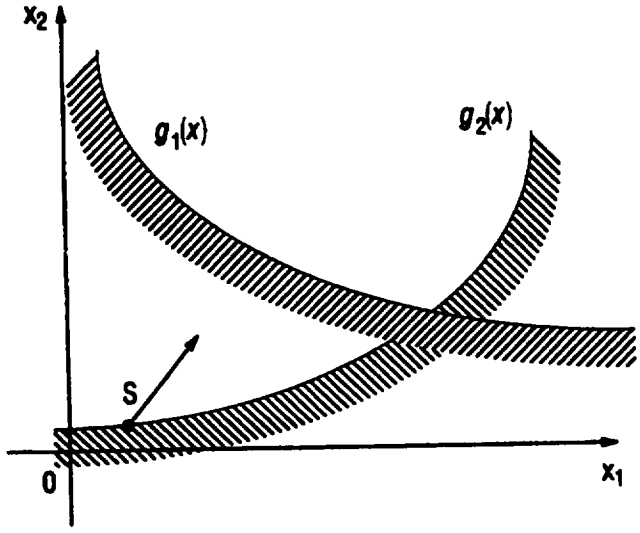


Fig.6.6 Violation of Constraint(s)

The new direction finding problem is now:

Find the search direction S and critical variable W that minimizes

$$\nabla F(\vec{X}^{q-1}) \cdot \vec{S} - \psi W \tag{6.33}$$

subject to

$$\nabla g_j(\vec{X}^{q-1}) \cdot \vec{S}^q + \theta_j W \leq 0 \qquad j \in J \tag{6.34}$$

$$\vec{S}^q \cdot \vec{S}^q + W \leq 1 \tag{6.35}$$

For discussion, assume the parameter θ_j in Eq. (6.34) is equal to 1.0, and the parameter W is a very large positive number. Then the second term dominates the minimizing of the function defined by Eq. (6.33). Therefore, any increase in the variable W forces the objective function to be more and more negative (i.e., it reduces the objective of the direction finding problem). However for W to increase the first term in Eq. (6.34) must become more and more negative. Since $\vec{S}^q$ is bounded by Eq. (6.35), the cosine of the angle between $\nabla g_j(\vec{X}^{q-1})$ and $\vec{S}^q$ must be moved closer and closer to -1.0 . For this to happen, $\vec{S}^q$ must point in a direction opposite to $g_j(\vec{X}^{q-1})$. That is $\vec{S}^q$ must point straight back toward the feasible region. The first term in Eq.(6.33) is included simply as a means to reduce the true objective function if possible while seeking a feasible design.

Now consider the value of θ_j in Eq.(6.34). This is referred to as a push-off factor since its value determines how hard to push away from this violated constraint. If $\theta_i = 0$, then even increasing W does not require a move anywhere except tangent to the constraint. Also this move will probably not move the design back to the feasible region. Therefore some positive value is needed. In optimizer, the more the constraint is violated, the greater the push-off away from the violated constraint. Finally, the value of ψ is initially chosen as a small number, such as 5.0.

<u>One dimensional search</u>

Having determined a usable-feasible search direction, the problem now becomes one of determining how far the optimization can move in that direction. A variety of possibilities exist depending on the starting point X^{q-1}. However, in each case, the optimization program makes use of polynomial approximations to the objective and constrained functions in the one-

dimensional direction defined by $\vec{S}$. The basic concept is to try some initial value for α^* in Eq.(6.25) and evaluate the corresponding objective and constraint functions. At the beginning of the optimization process, very little information is available except that the function values and their derivatives with respect to α^* are available. However, we consider the objective function (the same algebra applies to constraints) and create a first order approximation to $F(\alpha^*)$ in terms of α^*.

$$F(\vec{X}^q) = F(\vec{X}^{q-1} + \alpha^* \vec{S}^q) \tag{6.36}$$

Thus, a linear approximation to $F(\vec{X}^q)$ is

$$F(\vec{X}^q) = F(\vec{X}^{q-1}) + \left\{ \sum_{j=1}^{n} \left[\frac{\partial F(\vec{X}^{q-1})}{\partial X_i} \right] \cdot \left[\frac{\partial X_i}{\partial \alpha^*} \right] \right\} \cdot \alpha^* \tag{6.37}$$

or

$$F(\vec{X}^q) = F(\vec{X}^{q-1}) + \left[\frac{dF(\vec{X}^{q-1})}{d\alpha^*} \right] \cdot \alpha^* \tag{6.38}$$

But $\left[\frac{\partial F(\vec{X}^{q-1})}{\partial X_i} \right]$ is just the j-th entry of $\nabla F(\vec{X}^{q-1})$, or $[\nabla F(\vec{X}^{q-1})]_i$.
Also from Eq.(6.25), $\left[\frac{\partial X_i}{\partial \alpha^*} \right] = S_i$. Therefore,

$$\frac{dF(\vec{X}^{q-1})}{d\alpha^*} = \nabla F(\vec{X}^{q-1}) \cdot \vec{S}^q \tag{6.39}$$

Since both terms in Eq.(6.37) are available, we have the slope of function at $\alpha^* = 0$ for any function (objective or constraint) for which the gradient is available.

Now consider how this information might be used. Since this is the first step in the optimization process, we may try to reduce the objective function by some fraction, for example 10%, which can be stated as

$$F(\vec{X}^q) = F(\vec{X}^{q-1}) + \frac{dF(\vec{X}^{q-1})}{d\alpha^*} \cdot \alpha^* = F(\vec{X}^{q-1}) - 0.1 \cdot \mid F(\vec{X}^{q-1}) \mid \tag{6.40}$$

from which a proposed α^* is obtained as follows:

$$\alpha_{est}{}^* = -\frac{0.1 \cdot \mid F(\vec{X}^{q-1}) \mid}{\left[\frac{dF(\vec{X}^{q-1})}{d\alpha^*}\right]} \tag{6.41}$$

This results in an estimate of α^* which reduces the objective function by 10% . However since the gradients of some constraints are probably available, the other possible moves can be calculated as well. Remember that Eq.(6.38) applies to a constraint by simply substituting the constraint gradient for the objective gradient. Now assume that some constraint gradients for the constraints exist that are not critical, and it is desired to estimate how far to move to make one of them critical. That is, instead of reducing the value of the function by 10%, as was done for the objective, it is desired to estimate a move in α^* that places it near the constraint boundary. Thus, a linear approximation to find $g_j(\vec{X}^q) = 0$, is

$$g_j(\vec{X}^q) = g_j(\vec{X}^{q-1}) + \left[\frac{g_j(\vec{X}^{q-1})}{d\alpha^*}\right] \cdot \alpha^* = 0 \tag{6.42}$$

and an estimate for α^* is

$$\alpha_{est}{}^* = -\frac{g_j(\vec{X}^{q-1})}{\left[\frac{dg_j(\vec{X}^{q-1})}{d\alpha^*}\right]} \tag{6.43}$$

Therefore, even at the beginning of the one dimensional search a considerable amount of information is available to direct the process. Using the estimates for α^* given by Eq.(6.41) and Eq.(6.43) for each non-critical constraint, the smallest positive proposed α^* is taken as the first estimate of how far to move.

If constraints are currently violated, Eq.(6.43) can still be used. Applying this approximation to all violated constraints, the largest proposed value of α^* is selected as an estimate of how far to move to overcome all constraint violations. If the projected move to a constraint that is not currently active is smaller than this, then only move to that constraint as a first estimate since we do not want to add new constraints to the set of violated constraints.

Using similar approximations, we can also estimate an upper bound on α^* that forces all design variables to their lower or upper bounds. Then this provides a maximum allowable value for α^*.

The one-dimensional search process now proceeds to find the bounds on the α^* that contain the solution. Once the bounds on α^* are known, the constrained minimum is found by interpolation. Since $\vec{S}^q$ has been chosen as a direction of improving the design, the search can be limited to positive values of α^*. At this point, the best search direction to improve the design is determined and is conducted in that direction.

Convergence to the Optimum

Since numerical optimization is an iterative process, one of the most critical and difficult task is determining when to stop. It is important to remember that the process described in this section only relates to the solution of the approximate optimization problem. The number of cycles through the entire design process is controlled similar criteria.

The first criterion requires that the relative change in the objective between iterations is less than a specified tolerance. Thus the criterion is satisfied if:

$$\frac{| F(\vec{X}^q) - F(\vec{X}^{q-1}) |}{| F(\vec{X}^{q-1}) |} \leq 0.001 \tag{6.44}$$

The second criterion is that the absolute change in the objective between the iterations is less than a specified tolerance. This criterion is satisfied if

$$| F(\vec{X}^q) - F(\vec{X}^{q-1}) | \leq 0.0001 \tag{6.45}$$

6.3 Summary

This section has attempted to provide a brief overview of the computational details of the optimization process. The method described here is known as the modified method of feasible directions. The one-dimensional search problem is generally referred to as the polynomial interpolation with bounds. The algorithm presented here is available as the software package

DOT (Design Optimization Tools) [3]. Several structural problems were optimized using the algorithms presented in this report. The examples include frame, plate and truss structures with stress, displacement, and frequency constraints. Multiple load conditions and constraints are considered[4-6]. The efficiency of using adaptive approximations in reliability analysis and optimization are well documented in the published papers [7-9]. Hence they are simply referenced.

Also, this work developed a Graphical User Interface which simplifies and helps the selection of random variables, distributions and solution strategies. Again, results are available in published papers[10]. The procedure was demonstrated on a turbine blade problem.

*

References

[1] Vanderplaats,G.N., Numerical Optimization Techniques for Engineering With Applications, McGraw-Hill, 1984.

[2] Moore, G.J., MSC/NASTRAN User's Guide on Design Sensitivity Analysis and Optimization, Vol. 67, 1992.

[3] VMA Engineering, DOT Users Manual, Version 4.0, 1993.

[4] Reddy, M.V., Grandhi, R.V., and Hopkins, D.A., "Reliability Based Structural Optimization: A Simplified Safety Index Approach," *Journal of Computers & Structures,* Vol. 53, No. 6, 1994, pp. 1407-1418.

[5] Chandu, S.V.L., and Grandhi, R.V., "General purpose Procedure for Reliability Based Structural Optimization Under Parametric Uncertainties," *Journal of Advances in Engineering Software,* Vol. 23, No.1, 1995, pp. 7-14.

[6] Luo, X., and Grandhi, R.V., "ASTROS for Reliability-Based Multidisciplinary Structural Analysis and Optimization," *Journal of Computers and Structures,* Vol. 62, No. 4, 1997, pp. 737.

[7] Wang, L., and Grandhi, R.V., "Intervening Variables and Constraint Approximations in Safety Index and probability Calculations," *Journal of Structural Optimization,* Vol. 10, No. 1, 1995, pp. 2-8.

[8] Wang, L., Grandhi, R.V., and Hopkins, D.A. "Structural Reliability Optimization Using An Efficient Safety Index Calculation Procedure," *International Journal for Numerical Methods in Engineering,* Vol. 38, No. 10, 1995, pp. 1721-1738.

[9] Grandhi, R.V., and Wang, L., "Reliability-Based Structural Optimization Using Improved Two-point Adaptive Nonlinear Approximations," *Journal of Finite Element Analysis and Design,* (accepted for publication), 1997.

[10] Rajagopalan, H., and Grandhi, R.V., "Reliability Based Structural Analysis and Optimization in X Windows Environment," *Journal of Computers and Structures,* Vol. 60, No. 1, 1996, pp. 1-20.

Appendix A: Multivariate Hermite Approximation

In order to demonstrate that the proposed Hermite function possesses the same value and the same derivatives at each of the p data points as the exact function, rewriting Eq. (2.5.3a), $h_i(S)$ at the t^{th} known point S_t can be given as

$$h_i(S_t) = \frac{(S_t - S_1)^T(S_i - S_1)}{(S_i - S_1)^T(S_i - S_1)} \cdot \frac{(S_t - S_2)^T(S_i - S_2)}{(S_i - S_2)^T(S_i - S_2)} \cdots \frac{(S_t - S_j)^T(S_i - S_j)}{(S_i - S_j)^T(S_i - S_j)} \cdots$$
$$\frac{(S_t - S_p)^T(S_i - S_p)}{(S_i - S_p)^T(S_i - S_p)}, \qquad j \neq i \tag{A.1}$$

Using this equation, it is easy to prove that $h_i(S_t) = 1$ when $t = i$ because the numerator is the same as the denominator for this case. When t is not equal to i, t must be the same as one of j values (j=1,2,...p, $j \neq i$) because j is also not equal to i. So a zero term appears in the product terms of Eq. (A.1) so that $h_i(S_t)$ equals zero. Therefore, $h_i(S)$ at the known data points S_t (t=1,2,...p) satisfy

$$h_i(S_t) = \begin{cases} 1 & t = i \\ 0 & t \neq i \end{cases}$$
$$i = 1, 2, ..., p, \qquad t = 1, 2, ..., p \tag{A.2}$$

Based on this equation and Eq. (2.5.3c), it is esy to prove that the derivatives of $h_i(S)$ at the known data points S_t (t=1,2,...,p) satisfy

$$\frac{\partial h_i(S_t)}{\partial s_k} = \begin{cases} \sum_{j=1, j\neq i}^{p} \frac{(s_{k,i} - s_{k,j})}{(S_t - S_j)^T(S_i - S_j)} & t = i \\ 0 & t \neq i \end{cases}$$
$$i = 1, 2, ..., p, \qquad t = 1, 2, ..., p \tag{A.3}$$

Substituting Eqs. (A.2) and (A.3) into (2.5.4), the function value at the t^{th} known point can be written as

$$\tilde{f}(S_t) = \sum_{i=1}^{p} \{g(S_i) + [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)]^T(S_t - S_i)\} h_i^2(S_t) \tag{A.4a}$$

The summation can be expanded as two parts

$$\begin{aligned}\tilde{f}(S_t) &= \{g(S_t) + [\nabla g(S_t) - 2g(S_t) \nabla h_t(S_t)]^T \cdot 0\} h_t^2(S_t) \\ &\quad + \sum_{i=1, i\neq t}^{p} \{g(S_i) + [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)]^T (S_t - S_i)\} \cdot 0 \\ &= g(S_t) \end{aligned} \tag{A.4b}$$

Substituting Eqs. (A.2) and (A.3) into (2.5.5), the derivatives at the t^{th} known point can be written as

$$\begin{aligned}\nabla \tilde{f}(S_t) = \sum_{i=1}^{p} \Big\{ & 2h_i(S_t) \nabla h_i(S_t) \{g(S_i) + [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)]^T (S_t - S_i)\} \\ & + h_i^2(S_t)[\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)] \Big\} \end{aligned} \tag{A.5a}$$

Again, by the expanding the summation into two parts

$$\begin{aligned}\nabla \tilde{f}(S_t) &= 2h_t(S_t) \nabla h_t(S_t) \{g(S_t) + [\nabla g(S_t) - 2g(S_t) \nabla h_t(S_t)]^T \cdot 0\} \\ &\quad + h_t^2(S_t)[\nabla g(S_t) - 2g(S_t) \nabla h_t(S_t)] \\ &\quad + \sum_{i=1, i\neq t}^{p} \Big\{ 2 \cdot 0 \cdot 0 \cdot \{g(S_i) + [\nabla g(S_i) - 2g(S_i) \cdot \nabla h_i(S_i)]^T (S_t - S_i)\} \\ &\quad + 0 \cdot [\nabla g(S_i) - 2g(S_i) \nabla h_i(S_i)] \Big\} \\ &= 2h_t(S_t) \nabla h_t(S_t) g(S_t) + h_t^2(S_t) \nabla g(S_t) - 2h_t^2(S_t) \nabla h_t(S_t) g(S_t) \\ &= 2 \nabla h_t(S_t) g(S_t) + \nabla g(S_t) - 2 \nabla h_t(S_t) g(S_t) \\ &= \nabla g(S_t) \end{aligned} \tag{A.5b}$$

Eqs. (A.4) and (A.5) demonstrate that the proposed multivariate Hermite approximation possesses the same function values and derivatives at each of the known data points as the original information.

Appendix B: Parameters of the Distribution of a Random Variable

B.1 Expected Value

Expected value is sometimes called the mathematical expectation, or expectation or the mean value of the random variable.

For a discrete polulation, the mean is just the summation of all discrete values where each value is weighted for the probability of its occurence.

Let x be a random variable of the discrete type with jump points x_k and probabilities p_k. The expected value of the random variable can be expressed as

$$E(x) = \sum_{k=1}^{n} x_k p_k \qquad (B.1)$$

if the series is absolutely convergent.

For a continuous type, let $f_x(x)$ be the density function of random variable x. The expected value of the random variable can be expressed as

$$E(x) = \int_{-\infty}^{+\infty} x f_x(x) dx \qquad (B.2)$$

if the intergral $\int_{-\infty}^{+\infty} x f_x(x) dx$ is absolutely convergent.

B.2 Variance

The variance of a random variable X is the expected value of its squared deviation from its expected value μ, denoted by $Var(X)$ or σ^2, or the measure of dispersion of the random variable around its expected value.

Let X be a random variable of the discrete type with jump points x_k and probabilities p_k.

The variance of the random variable can be expressed as

$$Var(X) = E[(X - E(X))^2] = E[(X - \mu)^2]$$

B.3 Expected Value and Variance of Functions

B.3.1 Expected value of functions

Consider $y = g(x)$ as the single-valued function of x and x as a random of the discrete type with jump points x_k and probabilities p_k. The expected value of the random variable $g(x)$ can be expressed as

$$E(y) = E[g(x)] = \sum_{k=1}^{\infty} p_k g(x_k) \tag{B.3}$$

if the series is absolutely convergent.

Also, let x be a random variable of the continuous type with density function $f_x(x)$. The expected value of the random variable $g(x)$ can be expressed as

$$E(y) = E[g(x)] = \int_{-\infty}^{+\infty} g(x) f_x(x) dx \tag{B.4}$$

if the integral $\int_{-\infty}^{+\infty} g(x) f_x(x) dx$ is absolutely convergent.

B.3.2 Variance of linear functions

Given n random variable x_1, x_2, ..., x_n and a set of n constants c_1, c_2, ..., c_n, the linear function is expressed as

$$Z = \sum_{i=1}^{n} c_i x_i \tag{B.5}$$

The expected value of the function Z is

$$E(Z) = \sum_{i=1}^{n} c_i E(x_i) \tag{B.6}$$

The variance of Z is

$$Var(Z) = \sum_{i=1}^{n} c_i^2 \sigma_{x_i}^2 + \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j \rho_{ij} \sigma_{x_i} \sigma_{x_j}, \qquad i \neq j \tag{B.7}$$

where ρ_{ij} is the correlation coefficient of x_i and x_j. If n random variables are independent, then

$$Var(Z) = \sum_{i=1}^{n} c_i^2 \sigma_{x_i}^2 \tag{B.8}$$

Appendix C: Calculation of Coordinates η

A secant formula is used for calculating the coordinates η_a and η_b.

$$\eta_{i+2} = \eta_i - g_i \frac{\eta_i - \eta_{i+1}}{g_i - g_{i+1}} \qquad (C.1)$$

where g_i and g_{i+1} are the performance function values at Y_a or Y_b with η_i and η_{i+1}, respectively. By taking one data point, say Y_b, calculation of η_b is explained. η_a can be obtained using a similar procedure. For initial trial points, it is convenient to first examine $\eta_1 = \beta$, and the performance function value g_1 at the point $Y_b(k\beta,\ k\beta, ...,\ \eta_1)$ is calculated. The sign of g_1 is used for determining whether η is greater or smaller than β. If $g_1 > 0$, $\eta_2 = (1 + 0.5k^2)\beta$, otherwise, $\eta_2 = (1 - 0.5k^2)\beta$. With these two initial points, η_1 and η_2, the equation given in Eq. (C.1) is used to obtain η_3 and a new point $Y_b(k\beta,\ k\beta, ...,\ \eta_3)$. The performance function value g_3 at Y_b is calculated to check whether Y_b is on the limit state surface or not. If it is on the surface, stop the iteration; otherwise, continue using Eq. (C.1) until $g_{i+2}(Y_b)$=0.

Appendix D: Asymptotic Expansion of a Multinormal Integral

In Ref. [14] of Chapter 4 (this report), asymptotic expansions are given for integrals of the form

$$\hat{I}(\lambda) = \int_D exp(\lambda) f(U)) f_0(U) dU, \quad (\lambda \to \infty) \tag{D.1}$$

D is a fixed domain in the n-dimensional space, $f(U)$ and $f_0(U)$ are at least two times continuously differentiable functions. Further, it is assumed that the boundary of D is given by the point U with $h(U) = 0$, where $h(U)$ is also at least twice continuously differentiable. It is shown that, if $f(U)$ has no global maximum with respect to D at an interior point of D, the asymptotic behavior of $\hat{I}(\lambda)$ depends on the points on the boundary where $f(U)$ attains its global maximum on the boundary. Due to the Lagrange multiplier theorem, a necessary condition for these points is that $\nabla f(U) = K \cdot \nabla h(U)$ (K is a constant). The contribution of one of these points to the asymptotic expansion of $\hat{I}(\lambda)$ is given in Eqs. 8, 3, 64, page 340, Ref. [14] of Chapter 4 (this report). Defining $D = [U; g(U;1) < 0]$, $\lambda = \beta^2$, $f(U) = -|U|^2/2$, $h(U) = g(U;1)$, the formula can be applied to obtain an asymptotic expansion for $I(\beta)$. Due to the assumption made at the beginning, there is only one point U on the surface $g(U;1) = 0$ with minimal distance to the origin, i.e. only one point U_0 in which $-|U|^2/2$ achieves its maximum. Eqs. 8,3,64 yields then

$$I(\beta) \sim (2\pi)^{(n-1)/2} exp(\frac{-\beta^2 |U_0|^2}{2}) \beta^{-(n+1)} |J|^{-1/2}, \quad (\beta \to \infty) \tag{D.2}$$

with

$$J = \sum_{i=1}^{n} \sum_{j=1}^{n} U_0^i U_0^j cof(-\delta_{ij} - K g_{ij}) \tag{D.3}$$

where

$$U_0^i = ith \quad component \quad of \quad U_0$$

$$\delta_{ij} = \delta \quad Kronecker \quad Symbol$$

$$g_{ij} = \frac{\partial^2 g(U;1)}{\partial u_i \partial u_j}|_{U=U_0}$$

$$K = |\Delta g(U_0;1)|^{-1}$$

$cof(-\delta_{ij} - Kg_{ij})$ denotes the cofactor of the element $(-\delta_{ij} - Kg_{ij})$ in the matrix $(-\delta_{ij} - Kg_{ij})_{i,j=1,...,n}$. Since $|U_0| = 1$, the formula simplifies

$$I(\beta) \sim (2\pi)^{(n-1)/2} exp(\frac{-\beta^2}{2}) \beta^{-(n+1)} |J|^{-1/2} \tag{D.4}$$

Due to the rotational symmetry it can be assumed for further considerations, that $U_0 = (0,...,0,1)$ (i.e. the unit vector in the direction of the u_n-axis). Then, since U_0 is parallel to the gradient of $g(U;1)$ at U_0 due to the Lagrange multiplier theorem, the tangential space of the hyper surface at U_0 is spanned by the unit vectors in the direction of the first $n-1$ axes. Then J is given (using the definition of the cofactor):

$$|J| = |cof[-\delta_{nn} - Kg_{nn}]| = |det(B)| \tag{D.5}$$

with $B = (\delta_{1m} + Kg_{1m})_{1,m=1,...,n-1}$

Defining

$$\bar{D} = (-Kg_{1m})_{1,m=1,...,n-1}$$

and denoting the unity matrix by I:

$$det(B) = det(I - \bar{D}) \tag{D.6}$$

$Det(B)$ is given by the product of the eigenvalues of B, which are the roots of $det(B-\kappa I) = det((1-\kappa)I - \tilde{D})$. But these roots are given by $1-\kappa_i$ $(i = 1,...,n-1)$, in which the $\kappa_i^{'s}$ are the eigenvalues of the matrix D. This gives:

$$|J| = |\sum_{j=1}^{n-1}(1+\kappa_j)\beta| \tag{D.7}$$

These eigenvalues are the main curvatures of the surface at U_0. The curvature is defined positive. Eq. (D.7) shows, in which cases the approximation is not applicable. Since U_0 is a point on the surface with minimal distance to the origin, the main curvatures at U_0 must not be larger than unity, elsewhere, consider a point U on the surface near U_0 in the direction of a principal axis of curvature at U_0 with curvature κ_i larger than unity. Due to the definition of the curvature, the curve on the surface connecting U_0 and U is approximated by a part of a circle in the same direction through U_0 with radius $\frac{1}{\kappa_i}$ and center $(0,...,0,1-1/\kappa_i)$. Using elementary trigonometric relations, for small distances $|U-U_0|$ the squared distance of U to the origin is approximately

$$|U|^2 \approx 1 + \frac{(1-\kappa_i)|U-U_0|^2}{2} < 1 \tag{D.8}$$

This contradicts the assumption, that U_0 is a point on the surface with minimal distance to the origin with respect to the surface and therefore $\kappa_i \leq 1$. Due to this

$$|J| = \prod_{j=1}^{n-1}(1+\kappa_j\beta) \tag{D.9}$$

In the case that one curvature is exactly equal to unity, the approximation can not be used. Then it becomes necessary to study higher derivatives of $g(U)$ and the global behavior of the function.

Appendix E: Hermite and Laguerre Integral Parameters

E.1 Hermite Intergal Parameters

The Hermite integral formula can be used to solve the following integral

$$\int_{-\infty}^{\infty} f(x)dx = \sum_{i=1}^{N} \lambda_i e^{\xi_i^2} f(\xi_i) \qquad (E.1)$$

where N is the number of Hermite integral terms. The Hermite integral formula with ten terms is used, and its parameters are given in Table E1.

For the integration of Eq. (4.3.30), the ten-term Hermite formulas are given as

$$H^{(0)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}}\xi_i^m} \qquad (E.2a)$$

$$H^{(1)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a\xi_i^m \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}}\xi_i^m} \qquad (E.2b)$$

$$H^{(2)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^2\xi_i^{2m} \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}}\xi_i^m} \qquad (E.2c)$$

$$H^{(3)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^3\xi_i^{3m} \cdot e^{-\frac{1}{2}\xi_i^2 - \frac{a}{c_{0,1}}\xi_i^m} \qquad (E.2d)$$

For the integration of Eq. (4.3.36), the ten-term Hermite formulas are given as

$$\bar{H}^{(0)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot e^{-\frac{1}{2}\xi_i^2 + \frac{a}{c_{0,2}}\xi_i^m} \qquad (E.3a)$$

$$\bar{H}^{(1)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a\xi_i^m \cdot e^{-\frac{1}{2}\xi_i^2 + \frac{a}{c_{0,2}}\xi_i^m} \qquad (E.3b)$$

$$\bar{H}^{(2)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^2\xi_i^{2m} \cdot e^{-\frac{1}{2}\xi_i^2 + \frac{a}{c_{0,2}}\xi_i^m} \qquad (E.3c)$$

$$\bar{H}^{(3)} = \sum_{i=1}^{10} \lambda_i e^{\xi_i^2} \cdot a^3\xi_i^{3m} \cdot e^{-\frac{1}{2}\xi_i^2 + \frac{a}{c_{0,2}}\xi_i^m} \qquad (E.3d)$$

where λ_i and ξ_i are Hermite integral parameters which are given in Table E1.

Table E1. Hermite Integral Parameters

i	ξ_i	$\lambda_i e^{\xi_i^2}$
1	0.3429013272	0.6870818540
2	-0.3429013272	0.6870818540
3	1.0366108298	0.7032963231
4	-1.0366108298	0.7032963231
5	1.7566836493	0.7414419319
6	-1.7566836493	0.7414419319
7	2.5327316742	0.8206661264
8	-2.5327316742	0.8206661264
9	3.4361591188	1.0254516914
10	-3.4361591188	1.0254516914

Table E2. Laguerre Integral Parameters

i	$\bar{\xi}_i$	$\bar{\lambda}_i e^{\bar{\xi}_i}$
1	0.1523222277	0.3914311243
2	0.8072200227	0.9218050285
3	2.0051351556	1.4801279099
4	3.7834739733	2.0867708076
5	6.2049567779	2.7729213897
6	9.3729852517	3.5916260681
7	13.4662369111	4.6487660021
8	18.8335977890	6.2122754198
9	26.3740718909	9.3632182377

E. 2 Laguerre Integral Parameters

The Laguerre integral formula can be used to solve the following integral

$$\int_o^\infty f(x)dx = \sum_{i=1}^{N} \bar{\lambda}_i e^{\bar{\xi}_i} f(\bar{\xi}_i) \tag{E.4}$$

where N is the number of Laguerre integral terms. The Laguerre integral formula with nine terms is used, and its parameters are given in Table E2. For the integration of Eq. (4.3.48), the nine-term Laguerre formulas are given as

$$L^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(1)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2\bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(3)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3\bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,1}}\bar{\xi}_i^m} \tag{E.5a}$$

$$\bar{L}^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(1)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2\bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(3)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3\bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 - \frac{a}{c_{0,2}}\bar{\xi}_i^m} \tag{E.5b}$$

In the case that all a_i are negative, and $V_1 < 0$ and $V_2 > 0$, the integration of Eq. (4.3.49) is computed as

$$L^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(1)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2\bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m}$$

$$L^{(3)} = -\sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3\bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,1}}\bar{\xi}_i^m} \qquad (E.6a)$$

$$\bar{L}^{(0)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(1)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a\bar{\xi}_i^m \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(2)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^2\bar{\xi}_i^{2m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m}$$

$$\bar{L}^{(3)} = \sum_{i=1}^{9} \bar{\lambda}_i e^{\bar{\xi}_i} \cdot a^3\bar{\xi}_i^{3m} \cdot e^{-\frac{1}{2}\bar{\xi}_i^2 + \frac{a}{c_{0,2}}\bar{\xi}_i^m} \qquad (E.6b)$$

where $\bar{\lambda}_i$ and $\bar{\xi}_i$ are Laguerre integral parameters which are given in Table E2.