

AVALIAÇÃO DE FIABILIDADE ESTRUTURAL RECORRENDO A MÉTODOS DE SIMULAÇÃO

JOSÉ NUNO OLIVEIRA ALMEIDA ALVES DE MATOS

Dissertação submetida para satisfação parcial dos requisitos do grau de
MESTRE EM ENGENHARIA CIVIL — ESPECIALIZAÇÃO EM ESTRUTURAS

Orientador: Professor Doutor António Abel Ribeiro Henriques

JUNHO DE 2010

MESTRADO INTEGRADO EM ENGENHARIA CIVIL 2009/2010

DEPARTAMENTO DE ENGENHARIA CIVIL

Tel. +351-22-508 1901

Fax +351-22-508 1446

✉ miec@fe.up.pt

Editado por

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Rua Dr. Roberto Frias

4200-465 PORTO

Portugal

Tel. +351-22-508 1400

Fax +351-22-508 1440

✉ feup@fe.up.pt

🌐 <http://www.fe.up.pt>

Reproduções parciais deste documento serão autorizadas na condição que seja mencionado o Autor e feita referência a *Mestrado Integrado em Engenharia Civil - 2009/2010 - Departamento de Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, 2009.*

As opiniões e informações incluídas neste documento representam unicamente o ponto de vista do respectivo Autor, não podendo o Editor aceitar qualquer responsabilidade legal ou outra em relação a erros ou omissões que possam existir.

Este documento foi produzido a partir de versão electrónica fornecida pelo respectivo Autor.

À minha família

*"We learn something every day, and lots of times it's that what we learned the day before
was wrong."*

Bill Vaughan

AGRADECIMENTOS

A realização da presente dissertação envolveu, de forma directa e indirecta, a contribuição de diversas pessoas as quais não posso deixar de mostrar a minha gratidão.

Ao professor Abel Henriques pela empenhada orientação, colaboração, sugestões, conhecimentos transmitidos e incentivo no decurso da elaboração dos trabalhos desenvolvidos.

Ao meu amigo Ricardo Greenfield pela disponibilidade e a grande ajuda no funcionamento do “software” Matlab.

À Minha família pela dedicação, afecto, paciência e compreensão não só durante a fase de realização da presente dissertação, mas também ao longo de toda a minha vida.

À Filipa e aos meus amigos, pela amizade e companheirismo sempre reconfortantes.

RESUMO

Conceitos como fiabilidade e probabilidade, há muito que são mencionados nas metodologias de dimensionamento de estruturas. Teoricamente elementos iguais deveriam ter comportamentos semelhantes, mas como se sabe não existem elementos exactamente iguais, estando qualquer elemento sujeito a variações. Por exemplo, em ensaios a vigas metálicas idênticas haverá sempre múltiplos resultados. Entre estes resultados, alguns ocorrem com maior frequência que outros. Devido a esta incerteza, nas últimas três décadas, os códigos de dimensionamento têm vindo a ser modificados a fim de incorporar estes conceitos e melhorar o aproveitamento estrutural.

A segurança estrutural depende das incertezas associadas aos parâmetros de cálculo e das respectivas distribuições probabilísticas. Sendo a construção uma ciência dependente de inúmeras fontes de incerteza, é necessário considerar aspectos probabilísticos para avaliar de forma adequada o resultado esperado. Os métodos tradicionais encontraram a solução em metodologias determinísticas, usando coeficientes de segurança. Através destes coeficientes é possível dimensionar uma estrutura de um modo simples, garantindo segurança estrutural. Todavia estes métodos apresentam um contratempo, quando são introduzidas novas situações de dimensionamento, como por exemplo materiais novos, não existem coeficientes de segurança calibrados. Assim, é de grande interesse para a construção civil, desenvolver métodos que permitam dimensionar estruturas, sem recurso a coeficientes de segurança existentes. Recorrendo a métodos probabilísticos, não só é possível dimensionar uma situação estrutural nova, como também é possível calibrar um coeficiente de segurança para esta mesma situação.

Assim esta dissertação tem como principal objectivo desenvolver o conhecimento de métodos de simulação e elaborar uma ferramenta de apoio à avaliação de fiabilidade estrutural com recurso a estes métodos, permitindo a incorporação das incertezas associadas aos parâmetros de cálculo através das respectivas distribuições. Para a elaboração do programa de apoio, serão utilizados dois métodos de simulação, o método Monte Carlo que é caracterizado pela total aleatoriedade nas combinações de variáveis e o método do Hipercubo Latino que é baseado no anterior mas aplicando uma técnica de amostragem estratificada.

O programa desenvolvido está dividido em dois módulos. O primeiro módulo recebe variáveis aleatórias e através dos dois métodos de simulação, atrás referidos, gera combinações aleatórias de todas as variáveis envolvidas. Neste ponto o utilizador recebe uma lista com todas as amostras e procede ao tratamento das mesmas. O segundo módulo é usado quando os resultados estiverem prontos para ser tratados. O utilizador introduz uma lista com resultados e o programa devolve o índice de fiabilidade e a probabilidade de rotura.

Esta dissertação vai começar por desenvolver os conceitos básicos de estatística e depois aprofunda o conhecimento da fiabilidade estrutural. De seguida desenvolvem-se os dois métodos de simulação e aplicam-se a um exemplo simples. Com estes três conhecimentos assimilados expõe-se a estrutura do programa e os seus fundamentos. Na última parte da dissertação está disponível um caso prático de aplicação do programa em questão.

PALAVRAS-CHAVE: Fiabilidade estrutural, Técnicas de simulação, Betão armado, Flexão, Cálculo automático.

ABSTRACT

Concepts such as reliability and probability have long been referred in methodologies for design of structures. Theoretically, similar elements should have the same behaviour, but as we know no element is exactly alike, therefore being subject to variations. For example, when testing identical steel beams, there will be multiple outcomes. Among these outcomes there will be some that occur more frequently than others. Due to this uncertainty, in the last three decades, design codes have been modified to incorporate these concepts and improve structural use.

Structural safety depends on uncertainties linked to the calculations parameters and to probabilistic distribution. Knowing this and that the construction science has innumerable sources of uncertainties, it's necessary to incorporate probabilistic aspects to adequately assess the expected result. The traditional methods found the answer in deterministic methodologies, using safety coefficients. Through these coefficients it's possible to design structures in a simple and ensuring structural safety. However this method has a setback, when designing new structural situations, such as new construction techniques and new materials, the existing coefficients are inadequate. Therefore it's of great interest for the construction industry, to develop methods that don't rely on these safety coefficients. Using probabilistic methods not only is possible to design new structural situations, as it is possible to create an adequate coefficient for the same situation.

Thus, the main objective of this thesis is to develop the knowledge of simulation techniques and to prepare a tool that supports evaluations of structural reliability, through simulation methods. This way it's possible to incorporate the uncertainties linked to the calculations parameters in the structural design. In-built in the program, there will be two simulation techniques, the Monte Carlo method and the Latin Hypercube method. The first method is characterized by the complete randomness in the combinations of variables. The second one is a stratified sampling method that results from the previous one, by applying variance reduction techniques.

The software is divided into two modules. Using the two simulation methods, mentioned above, the first module generates random combinations of all variables involved in a structural problem. At this point, the user receives a list with all the samples and proceeds with the calculations. The second module is used when all the results are ready to be treated. The user enters a list of results and the program returns the reliability index and the probability of failure.

This thesis will start by developing the basics concepts of statistics and then deepens the understanding of structural reliability. Afterwards the thesis will develop knowledge about the two simulation methods and applies them to a simple practical case. In the latter part of the thesis there's an example of practical application for the developed software.

KEYWORDS: Structural Reliability, Simulation Techniques, Reinforced Concrete, Flexion, Automatic Calculation.

ÍNDICE GERAL

AGRADECIMENTOS	i
RESUMO	iii
ABSTRACT	v
1. APRESENTAÇÃO DA DISSERTAÇÃO	1
1.1. ENQUADRAMENTO	1
1.2. OBJECTIVOS PROPOSTOS	2
1.3. ESTRUTURA DA DISSERTAÇÃO	2
2. AVALIAÇÃO DE FIABILIDADE ESTRUTURAL	5
2.1. CONCEITOS DE ESTATÍSTICA	5
2.1.1. INTRODUÇÃO	5
2.1.2. CONCEITOS ESTATÍSTICOS UTILIZADOS	5
2.1.2.1. Variáveis aleatórias	5
2.1.2.2. Função de distribuição e função densidade de probabilidade	6
2.1.2.3. Parâmetros de distribuição	7
2.1.3. DISTRIBUIÇÕES DE PROBABILIDADE COMUNS EM ENGENHARIA CIVIL	8
2.1.3.1. Leis de probabilidade discretas	8
2.1.3.2. Leis de probabilidade contínuas	11
2.2. CONCEITOS DE SEGURANÇA DE ESTRUTURAL	16
2.2.1. INTRODUÇÃO	16
2.2.2. INCERTEZAS ASSOCIADAS À AVALIAÇÃO DE SEGURANÇA DE ESTRUTURAS	16
2.2.3. ESTADOS LIMITES	18
2.2.4. VERIFICAÇÃO DA SEGURANÇA AOS ESTADOS LIMITES	18
2.2.5. MEDIDORES DE FIABILIDADE ESTRUTURAL	19
2.2.5.1. Probabilidade de rotura	19
2.2.5.2. Índice de fiabilidade	22
2.2.6. COMPARAÇÃO ENTRE OS MÉTODOS DE FIABILIDADE E OS MÉTODOS DE SIMULAÇÃO	25
2.3. MÉTODOS DE SIMULAÇÃO	26
2.3.1. INTRODUÇÃO	26
2.3.2. MÉTODO DE MONTE CARLO	26
2.3.2.1. Geração de valores aleatórios	27

2.3.2.2. Teste de Hipóteses.....	29
2.3.2.3. Cálculo da variância, Coeficiente de variação e do erro.....	31
2.3.3. TÉCNICAS DE REDUÇÃO DE VARIÂNCIA – MÉTODO DO HIPERCUBO LATINO	32
2.3.3.1. Geração de valores aleatórios.....	33

3. PROGRAMAÇÃO..... 37

3.1. INTRODUÇÃO	37
3.2. PRIMEIRO MÓDULO – SIMULAÇÃO DE AMOSTRAS.....	39
3.3. SEGUNDO MÓDULO – AVALIAÇÃO DA FIABILIDADE ESTRUTURAL	42
3.4. EXEMPLO DE APLICAÇÃO	44
3.4.1. GEOMETRIA E CARACTERIZAÇÃO DAS VARIÁVEIS ALEATÓRIAS	44
3.4.2. SIMULAÇÃO DE AMOSTRAS.....	47
3.4.3. AVALIAÇÃO DE FIABILIDADE ESTRUTURAL.....	48
3.4.4. SIMULAÇÃO E AVALIAÇÃO DE FIABILIDADE ESTRUTURAL – MÉTODO DE MONTE CARLO	50
3.4.5. SIMULAÇÃO E AVALIAÇÃO DE FIABILIDADE ESTRUTURAL – MÉTODO DO HIPERCUBO LATINO.....	54

4. CASO PRÁTICO

4.1. INTRODUÇÃO	59
4.2. GEOMETRIA, MATERIAIS E ACÇÕES	59
4.3. AVALIAÇÃO DE FIABILIDADE ESTRUTURAL	64
4.3.1. SIMULAÇÃO DE 10000 AMOSTRAS PELO MÉTODO DE MONTE CARLO.....	64
4.3.2. SIMULAÇÃO DE 100000 AMOSTRAS PELO MÉTODO DE MONTE CARLO.....	67
4.3.3. SIMULAÇÃO DE 100 AMOSTRAS PELO MÉTODO DO HIPERCUBO LATINO	71
4.3.4. COMENTÁRIOS FINAIS	74
4.4. ALTERAÇÃO DAS CARACTERÍSTICAS DA PASSAGEM SUPERIOR.....	76

5. CONCLUSÃO..... 81

5.1. CONCLUSÕES FINAIS.....	81
5.2. DESENVOLVIMENTOS FUTUROS.....	82

6. BIBLIOGRAFIA

ANEXOS

ÍNDICE DE FIGURAS

Fig. 2.1 - Variáveis aleatórias.....	5
Fig. 2.2 - Função de densidade de Probabilidade (FDP) e Função de Distribuição (FD) (adaptado de Haldar & Mahadevan, 2000).....	7
Fig. 2.3- Representação gráfica da função densidade de probabilidade gausseana.	11
Fig. 2.4 - Representação gráfica da função de densidade de probabilidade conjunta Parte I (adaptado de Haldar & Mahadevan, 2000).....	20
Fig. 2.5 - Representação gráfica da função de densidade de probabilidade conjunta Parte II (adaptado de Delgado, 2002).....	20
Fig. 2.6 – Representação gráfica das funções f_R e f_S , simplificada.....	21
Fig. 2.7 - Representação gráfica da relação do índice de fiabilidade com a probabilidade de rotura (adaptado de Delgado, 2002).....	23
Fig. 2.8 - Fases do método de Monte Carlo.....	27
Fig. 2.9 - Representação gráfica da transformação inversa (adaptado de Veiga, 2008).....	28
Fig. 2.10 - Histograma com curva teórica ajusta (à esquerda); curva de frequências acumuladas com curva teórica ajustada (à direita).	29
Fig. 2.11 - Representação gráfica da primeira fase do método do Hipercubo Latino.....	33
Fig. 2.12 - Representação gráfica da segunda fase do método do Hipercubo Latino.....	34
Fig. 3.1 - Interface gráfica do programa de apoio à avaliação de fiabilidade estrutural.....	38
Fig. 3.2 - Imagem retirada do programa.....	39
Fig. 3.3 - Caracterização do parâmetro K das leis do tipo II (à esquerda) e III (à direita).....	40
Fig. 3.4 - Exemplo de caixa de mensagem com informação sobre os resultados dos testes de hipótese.....	41
Fig. 3.5 - Exemplo de caixa de mensagem com informação sobre os resultados dos testes de hipótese.....	41
Fig. 3.6 - Exemplo de mensagem de verificação dos testes de hipótese do segundo módulo.....	43
Fig. 3.7 - Interface gráfica com representação do histograma.....	44
Fig. 3.8 - Geometria da viga.....	45
Fig. 3.9 - Representação gráfica do equilíbrio de forças.....	45
Fig. 3.10 - Representação gráfica da equação do momento resistente em relação à face superior.....	47
Fig. 3.11 - Histograma com curva teórica ajustada do momento resistente para 100 amostras.....	49
Fig. 3.12 - Histogramas com curva teórica ajustada dos momentos actuantes para 100 amostras, quando $q=100\text{kN/m}$ (à esquerda) e $q=130\text{kN/m}$ (à direita).....	50
Fig. 3.13 - Histogramas com curva teórica ajustada das respostas estruturais para 100 amostras, quando $q=100\text{kN/m}$ (à esquerda) e $q=130\text{kN/m}$ (à direita).....	50
Fig. 3.14 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 1000 amostras.....	53
Fig. 3.15 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 5000 amostras.....	54

Fig. 3.16 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 10000 amostras.....	54
Fig. 3.17 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 100 amostras, pelo método do Hipercubo Latino.....	57
Fig. 3.18 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 1000 amostras, pelo método do Hipercubo Latino.....	57
Fig. 3.19 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 5000 amostras, pelo método do Hipercubo Latino.....	58
Fig. 3.20 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 10000 amostras, pelo método do Hipercubo Latino.....	58
Fig. 4.1 - Modelo estrutural da passagem superior.....	60
Fig. 4.2 - Secção a meio vão (aos 20 metros).....	60
Fig. 4.3 - Secção onde se dá o encontro com os pilares (aos 10 e 30 metros).....	60
Fig. 4.4 - Traçado do cabo de pré-esforço equivalente.....	61
Fig. 4.5 - Esquema de cargas aplicadas para a combinação 1	63
Fig. 4.6 - Esquema de cargas aplicadas para a combinação 2	63
Fig. 4.7 - Histogramas com curva teórica ajustada dos momentos resistentes para 10000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	66
Fig. 4.8 - Histogramas com curva teórica ajustada dos momentos actuantes para 10000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	67
Fig. 4.9 - Histogramas com curva teórica ajustada das respostas estruturais para 10000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	67
Fig. 4.10 - Histogramas com curva teórica ajustada dos momentos resistentes para 100000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	70
Fig. 4.11 - Histogramas com curva teórica ajustada dos momentos actuantes para 100000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	70
Fig. 4.12 - Histogramas com curva teórica ajustada das respostas estruturais para 100000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	71
Fig. 4.13 - Histogramas com curva teórica ajustada dos momentos resistentes para 100 amostras pelo método de Hipercubo Latino (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	73
Fig. 4.14 - Histogramas com curva teórica ajustada dos momentos actuantes para 100 amostras pelo método de Hipercubo Latino (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	73
Fig. 4.15 - Histogramas com curva teórica ajustada das respostas estruturais para 100 amostras pelo método de Hipercubo Latino (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)	74

Fig. 4.16 - Representação gráfica do cálculo do momento actuante total78

ÍNDICE DE TABELAS

Tabela 2.1 - Relação entre a probabilidade de rotura e o índice de fiabilidade.....	23
Tabela 2.2 - Valores Mínimos recomendados para o índice de fiabilidade (EC-0, 2002).	24
Tabela 2.3 - Probabilidade do afastamento máximo entre $F^*(x)$ e $F(x)$ devido aos factores aleatórios seja não inferior ao valor observado.	31
Tabela 2.4 – Resultado da simulação efectuada na figura 2.12.....	35
Tabela 3.1 - Exemplo de tabela exportada para a folha "Dados_Matlab".....	41
Tabela 3.2 - Exemplo de uma tabela exportada para a folha "Testes_de_Hipóteses"	42
Tabela 3.3 - Exemplo de resultados exportados para a folha "Resultados_Matlab" ,no segundo módulo	43
Tabela 3.4 - Parâmetros de distribuição das variáveis envolvidas	46
Tabela 3.5 - Resultados da simulação e dos testes de hipótese, para 100 amostras	47
Tabela 3.6 - Momentos e resposta estrutural para 100 amostras.....	48
Tabela 3.7 - Resultados da amostragem de 100 simulações	48
Tabela 3.8 - Resultados dos testes de Hipóteses para 1000 amostras	50
Tabela 3.9 - Resultados dos testes de Hipóteses para 5000 amostras	51
Tabela 3.10 - Resultados dos testes de Hipóteses para 10000 amostras	51
Tabela 3.11 – Momento resistente e momentos actuantes para 100, 1000, 5000 e 10000 simulações..	51
Tabela 3.12 - Resultados dos testes de hipótese no segundo módulo, para 1000, 5000 e 10000 amostras	52
Tabela 3.13 - Resposta estrutural para todas as amostragens.....	52
Tabela 3.14 - Resultados finais para 100, 1000, 5000,10000.....	53
Tabela 3.15 - Resultados dos momentos resistentes e momentos actuantes das amostras simuladas pelo método do Hiper cubo Latino.....	55
Tabela 3.16 - Resultados dos testes de hipóteses das amostras simuladas pelo método do Hiper cubo Latino	55
Tabela 3.17 - Resposta estrutural para o método do Hiper cubo Latino	56
Tabela 3.18 - Resultados finais para o método do Hiper cubo Latino	56
Tabela 4.1 - Parâmetros de distribuição das variáveis envolvidas (dados geométricos).....	61
Tabela 4.2 - Parâmetros de distribuição das variáveis envolvidas (betão, aço ordinário e aço de pré-esforço).....	62
Tabela 4.3 - Acções actuantes na passagem superior.....	62
Tabela 4.5 - Resultados dos testes de Hipótese das variáveis simuladas (Monte Carlo, 10000 amostras)	64
Tabela 4.6 – Média, desvio padrão, mínimos, máximos e resultados dos testes de Hipótese dos momentos resistentes e actuantes (Monte Carlo, 10000 amostras).....	65
Tabela 4.7 – Média, desvio padrão, mínimo e máximo da resposta estrutural (Monte Carlo, 10000 amostras)	66

Tabela 4.8 – Resultados finais da avaliação de fiabilidade estrutural (Monte Carlo, 10000 amostras)	66
Tabela 4.9 - Resultados dos testes de Hipótese das variáveis simuladas (Monte Carlo, 100000 amostras).....	68
Tabela 4.10 - Média, desvio padrão, mínimos, máximos e resultados dos testes de Hipótese dos momentos resistentes e actuantes (Monte Carlo, 100000 amostras)	68
Tabela 4.11 - Média, desvio padrão, mínimo e máximo da resposta estrutural (Monte Carlo, 100000 amostras).....	69
Tabela 4.12 - Resultados finais da avaliação de fiabilidade estrutural (Monte Carlo, 100000 amostras)	69
Tabela 4.14 - Média, desvio padrão, mínimo e máximo da resposta estrutural (Hiper-cubo Latino, 100 amostras).....	72
Tabela 4.15 - Resultados finais da avaliação de fiabilidade estrutural (Hiper-cubo Latino, 100 amostras)	73
Tabela 4.16 - Resumo das respostas estruturais obtidas nas simulações efectuadas.....	75
Tabela 4.18 - Resumo dos índices de fiabilidade e da probabilidade de rotura obtidos para todas as simulações	76
Tabela 4.19 - Armaduras deterioradas.....	77
Tabela 4.20 - Momentos resistentes e actuantes (Secções deterioradas).....	79
Tabela 4.21 - Média, desvio padrão, mínimo e máximo da resposta estrutural (secções deterioradas)	79
Tabela 4.22 - Resultados finais da avaliação de fiabilidade estrutural (secções deterioradas).....	80

SÍMBOLOS E ABREVIATURAS

- α – Parâmetro de distribuição da lei extremos do tipo I
- β – Índice de fiabilidade [adim.]
- $\beta(a, b)$ – Lei Beta
- γ – Constante de Euler [adim.]
- γ_1 – Coeficiente de assimetria [adim.]
- γ_2 – Coeficiente de achatamento [adim.]
- ε – erro [%]
- λ – Parâmetro de distribuição da lei de Poisson e lognormal [adim.]
- μ – Média
- μ_3 – Desvio cúbico
- μ_4 – Desvio quártico
- σ – Desvio padrão
- ξ – Parâmetro de distribuição da lei lognormal
- $\Gamma()$ – Função gama
- ϕ – diâmetro [mm]
- $\Phi()$ – Lei normal padronizada
- a – vários
- A_s – armadura de tracção [cm^2]
- A_p – armadura de tracção de pré-esforço [cm^2]
- $B(n, p)$ – Lei binomial
- b – vários
- c – recobrimento [m]
- COV – Coeficiente de variação calculado segundo Haldar & Mahadevan, 2000
- CV – Coeficiente de variação
- d – vários
- D_n – Máxima diferença entre as frequências acumuladas e a lei probabilística teórica
- \tilde{D}_n – Coeficiente de distribuição
- E-I(μ, α) - Lei de extremos tipo I
- E-II(μ, α) - Lei de extremos tipo II
- E-III(μ, α) - Lei de extremos tipo III
- F - força [kN]

f_c – tensão de compressão máxima do betão [MPa]

f_{sy} – tensão de cedência do aço [MPa]

f_{pu} - Tensão de rotura da armadura de pré-esforço [MPa]

$f_{p0,1}$ - Tensão limite de proporcionalidade a 0.1% da armadura de pré-esforço [MPa]

$F_X(x)$ – Função de distribuição

$F_X^{-1}(x)$ – Função de distribuição inversa

$f_X(x)$ – Função densidade de probabilidade

$f_{R,S}$ – Função densidade de probabilidade conjunta

$g(\)$ – Função de estado limite

i – Contador de ciclos

j – Contador de ciclos

K – Paramento de forma para as leis de extremos II e III, quando aplicados ao programa de apoio [adim.]

k – vários

L – comprimento do vão[m]

$LN(\lambda, \xi)$

M – momento [kN.m]

$N(\mu, \sigma)$ – Lei normal

p – Probabilidade

PP – Peso próprio [kN/m]

$Prev$ – Peso dos revestimentos e guarda corpos [kN/m]

Psc – Peso da sobrecarga [kN/m]

q – Carga distribuída [kN/m]

pf – Probabilidade de rotura

R – Capacidade resistente [kN.m]

S – Acção [kN.m]

U – Variável aleatória do método de Box-Müller

X – Variáveis aleatórias

Y – Variáveis aleatórias

Z – Resposta Estrutural [kN.m]

CEN - Comité Europeu de Normalização

EC-0 – Euro-Código 0

EC-1 – Euro-Código 1

ELU – Esta Limite Último

ELS – Esta Limite de Serviço

ELF – Esta Limite de Fadiga

FD – Função de distribuição

FDP – Função densidade de probabilidade

FOSM – First order second moment reliability method

FORM – First order reliability method

K-S – Teste De Kolmogorov-Smirnov

MC – Método de Monte Carlo

Norm. – Teste de Normalidade

RC – Reliability class

RSA – Regulamento de segurança e acções para estruturas de edifícios e pontes

SORM – Second order reliability method

1

APRESENTAÇÃO DA DISSERTAÇÃO

1.1. ENQUADRAMENTO

A existência de incertezas no âmbito da engenharia civil sempre foi reconhecida pelos engenheiros. Sendo uma das principais preocupações, atingir a fiabilidade estrutural minimizando os custos, adoptou-se um método determinístico que contabiliza as incertezas através do uso de coeficientes de segurança. A introdução de métodos probabilísticos permitiu obter coeficientes de segurança de forma uniformizada, uma vez que são definidos para a mesma probabilidade de falha. Coeficientes de segurança bem calibrados permitem dimensionar uma estrutura de modo simples, garantindo um bom aproveitamento estrutural. Como estes coeficientes são calibrados para as estruturas e materiais existentes na altura em que se elaboram as normas regulamentares, a utilização destes, deixa de ser ideal, quando surgem novas situações de dimensionamento estrutural (técnicas construtivas ou materiais). Para estes casos a utilização dos métodos probabilísticos permite considerar directamente as variabilidades reais dos parâmetros estruturais. Actualmente recorre-se a métodos semi-probabilísticos para o dimensionamento de estruturas, onde as variáveis envolvidas são representadas pelos seus valores nominais e característicos, afectados por coeficientes de segurança parciais. Para obter coeficientes de segurança adequados a cada situação estrutural, recorre-se aos métodos probabilísticos.

Para poder empregar métodos probabilísticos é necessário efectuar centenas de cálculos, ou até milhares, dependendo da complexidade do sistema estrutural ou do erro máximo admitido. Como é óbvio, efectuar estes cálculos manualmente seria demasiado demorado e dispendioso. Daí a necessidade de desenvolver uma ferramenta de apoio que efectue estes cálculos com maior rapidez e precisão. O uso destes programas automáticos proporciona uma grande vantagem, a capacidade de prever situações reais como, modos de rotura e tipos de resposta a diferentes acções, em tempos de computação admissíveis. O usufruto destes programas auxiliares desencadeou um desenvolvimento de técnicas probabilísticas capazes de incorporar um grande número de variáveis aleatórias, atribuindo a cada uma, um vasto conjunto de amostras. Estas novas técnicas usam métodos estatísticos e estocásticos para simular o comportamento real das grandezas envolvidas.

Conceitos como probabilidade de rotura e índice de fiabilidade têm vindo a ser utilizados como quantificadores de fiabilidade estrutural pelo seu carácter racional. A introdução dos mesmos, nos códigos de dimensionamento, permitiu a criação de valores mínimos recomendados, pelos quais um engenheiro civil se pode guiar.

Desta forma a presente dissertação vai apresentar os conceitos estatísticos necessários para desenvolver técnicas probabilísticas e vai também estudar dois métodos de simulação estocástica. Sendo também seu objectivo, explicar o funcionamento de um programa automático de apoio à avaliação de segurança de elementos estruturais.

1.2. OBJECTIVOS PROPOSTOS

O principal objectivo desta dissertação é elaborar uma ferramenta automática que apoie qualquer avaliação de segurança de elementos estruturais, permitindo a incorporação das incertezas associadas aos parâmetros de cálculo através das respectivas distribuições.

Tendo em conta os objectivos propostos, a dissertação foi organizada nas seguintes etapas:

- Pesquisa bibliográfica sobre técnicas de simulação, modelos existentes, conceitos e metodologias de aplicação;
- Pesquisa bibliográfica sobre os conceitos básicos associadas às técnicas de simulação;
- Estudo sobre as técnicas de avaliação e segurança de estruturas de forma a obter bases para o desenvolvimento do programa de cálculo automático;
- Pesquisa bibliográfica sobre plataformas computacionais que permitam elaborar programas de cálculo automático;
- Escolha da plataforma computacional ideal e implementação do programa de cálculo automático;
- Elaboração do programa de cálculo automático empregando os conhecimentos adquiridos nas etapas antecedentes;
- Aplicação do programa num caso prático e recolha de dados necessários para análise de fiabilidade estrutural posterior;
- Análise dos dados obtidos e comparação dos métodos de simulação utilizados;

1.3. ESTRUTURA DA DISSERTAÇÃO

A presente dissertação encontra-se dividida em 5 capítulos onde se descrevem temas fundamentais para atingir os objectivos propostos. Seguidamente é feita uma descrição breve de cada capítulo.

Capítulo 1

Capítulo introdutório onde é apresentado o enquadramento e os objectivos principais desta dissertação. São ainda expostas as etapas necessárias para atingir os objectivos propostos.

Capítulo 2

Neste capítulo são desenvolvidos os conceitos teóricos necessários para a realização do programa de apoio. Inicia-se pelos conhecimentos básicos de estatística e probabilidade. Dá-se a definição de variáveis aleatórias e a definição das distribuições probabilísticas usadas com frequência no âmbito da engenharia civil. De seguida abordam-se os conceitos de avaliação de fiabilidade estrutural, começando pelos estados limites e pelas incertezas associadas dos parâmetros envolvidos nos sistemas estruturais. Desenvolve-se o conceito de fiabilidade estrutural, apoiada pela definição da probabilidade de rotura e índice de fiabilidade. Serão desenvolvidos os diferentes níveis dos métodos de verificação de segurança. Descrevem-se as principais diferenças entre os processos de avaliação e aborda-se as diferenças entre a execução de uma análise determinística e probabilística. Por último abordam-se os principais métodos de simulação, usados para a avaliação de fiabilidade estrutural, com particular atenção ao método de Monte Carlo e ao método do Hipercubo Latino. Serão realçadas as potencialidades e limitações de ambos os métodos. Descrevem-se ainda os métodos de verificação de simulação, testes de hipóteses.

Capítulo 3

Apresenta-se a plataforma usada para elaborar o programa de apoio à avaliação de fiabilidade estrutural, expondo a suas principais potencialidades. Serão desenvolvidos os dois módulos a realizar, descrevendo a estrutura base do programa e completando com um exemplo de aplicação simplificado.

Capítulo 4

Aplicação dos conhecimentos adquiridos, testando o programa elaborado com o exemplo sugerido. Os dois métodos de simulação, Monte Carlo e Hipercubo Latino, serão aplicados a um caso prático. Obtidos os valores finais procede-se a uma análise de resultados.

Capítulo 5

Apresenta-se uma conclusão final, realçando os aspectos mais importantes com comentários e conclusões relevantes.

2

AVALIAÇÃO DE FIABILIDADE ESTRUTURAL

2.1. CONCEITOS DE ESTATÍSTICA

2.1.1. INTRODUÇÃO

A estatística tem vindo a ser reconhecida como tópico de grande interesse na formação de engenheiros civis, sendo por isso cada vez mais incluída nos cursos superiores como disciplina autónoma ou integrada noutras disciplinas. O seu principal objectivo é recolher, compilar, analisar e interpretar dados. Visto que esta dissertação envolve vários conceitos e métodos estatísticos, este ponto refere-se a tudo o que será necessário aprender, para elaborar o programa de apoio à avaliação em causa.

2.1.2. CONCEITOS ESTATÍSTICOS UTILIZADOS

2.1.2.1. Variáveis aleatórias

Em quase todos os acontecimentos relacionados com a engenharia não se pode prever um resultado final com total exactidão. O registo destes acontecimentos, fornece múltiplos resultados tendo ou não um padrão de distribuição. São assim consideradas experiências aleatórias e utiliza-se o termo de variável aleatória para as expressar. O conjunto de valores que uma variável aleatória toma denomina-se de espaço amostral. Grande parte dos elementos envolvidos em sistemas de engenharia são incertos e devem ser considerados como variáveis aleatórias.

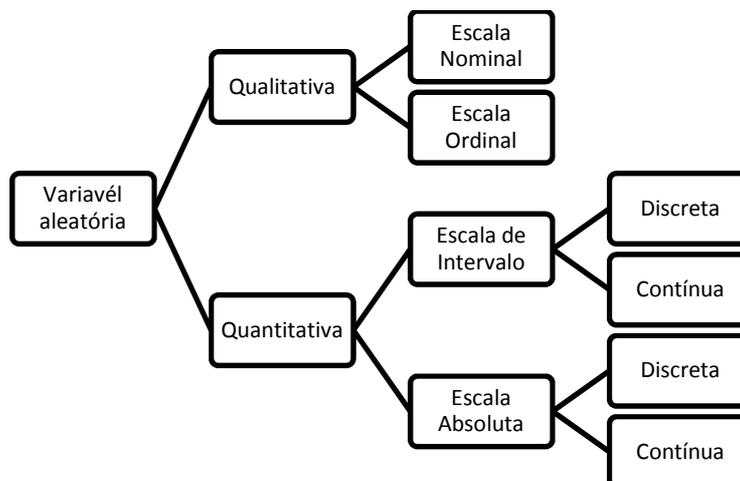


Fig. 2.1 - Variáveis aleatórias

As variáveis aleatórias podem ser classificadas em duas categorias diferentes (figura 2.1). A primeira categoria define a variável como qualitativa ou quantitativa, contudo nesta dissertação só serão abordadas as variáveis quantitativas. Por exemplo, quando se fala numa ciência exacta não se classifica um betão por razoável ou bom (qualitativa), mas sim por um escalão numérico (quantitativa). O segundo escalão distingue entre variáveis discretas e contínuas. A grande diferença entre os dois, está no género de número que cada uma pode assumir. As variáveis discretas só podem assumir um número finito, ou seja valores inteiros, enquanto as contínuas podem assumir valores numa escala contínua. Em problemas de engenharia estrutural a maioria das variáveis são contínuas.

2.1.2.2. Função de distribuição e função densidade de probabilidade

A função densidade de probabilidade, FDP, não fornece directamente informação sobre a probabilidade mas sim sobre a natureza da aleatoriedade e é representada por $f_X(x)$ (ver figura 2.2), em que X é uma variável aleatória e x é um determinado valor. Para calcular a probabilidade é necessário calcular a área entre dois pontos da curva. Através da seguinte expressão é possível calcular a probabilidade, P :

$$P(x_1 < X \leq x_2) = \int_{x_1}^{x_2} f_X(x) dx \quad (2.1)$$

Integrando a função densidade de probabilidade para um intervalo de $-\infty$ a x_2 obtêm-se a função de distribuição, FD, a função $F_X(x)$ (ver figura 2.2) que define a probabilidade de X ser inferior ou igual a x . Ao contrário da FDP, a FD não fornece dados sobre a natureza da aleatoriedade.

$$F_X(x) = P(X \leq x) = \int_{-\infty}^x f_X(x) dx \quad (2.2)$$

Frequentemente a única função disponível é a FD. Se for necessário analisar também a FDP utiliza-se a seguinte expressão, obtida a partir da primeira derivada da FD:

$$f_X(x) = \frac{dF_X(x)}{dx} \quad (2.3)$$

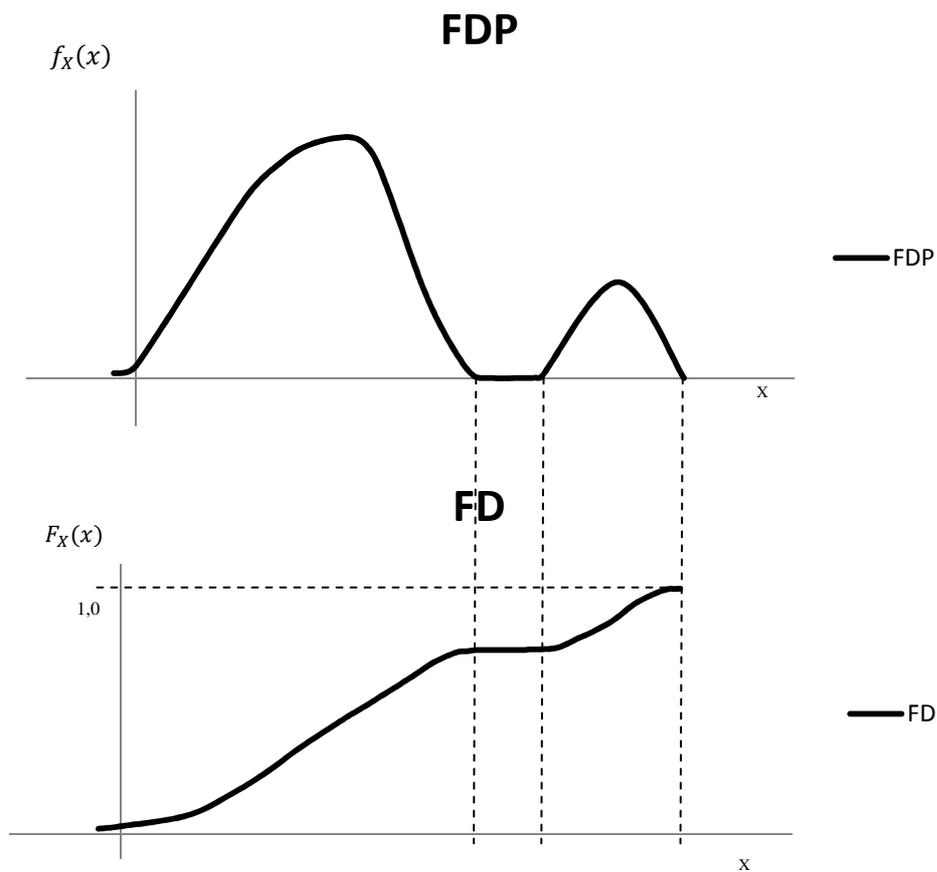


Fig. 2.2 - Função de densidade de Probabilidade (FDP) e Função de Distribuição (FD) (adaptado de Haldar & Mahadevan, 2000)

2.1.2.3. Parâmetros de distribuição

Os parâmetros de distribuição são um conjunto de definições que caracterizam as FD e FDP de uma variável. Por vezes basta conhecer dois dos parâmetros para conhecer a sua distribuição de probabilidade. Estes parâmetros são também conhecidos por momentos estatísticos. Como a maioria das variáveis usadas em engenharia são contínuas, só se vão mencionar os parâmetros que correspondem às mesmas.

Parâmetro de localização (média)

Valor esperado ou valor médio é designado por μ_X e é definido pela seguinte equação:

$$\mu_X = \int_{-\infty}^{+\infty} x \times f(x) dx \quad (2.4)$$

Parâmetros de dispersão

Apresenta-se seguidamente as definições que caracterizam a dispersão de distribuições de probabilidade.

Variância (momento de segunda ordem)

Variância de uma variável aleatória, X , é designada por σ_X^2 e define-se pela seguinte expressão:

$$\sigma_x^2 = \int_{-\infty}^{+\infty} (x - \mu_x)^2 \times f(x) dx \quad (2.5)$$

Desvio padrão (momento de segunda ordem)

O desvio padrão de uma variável aleatória X é designado por σ_x e obtém-se pela raiz quadrada da variância:

$$\sigma_x = \sqrt{\sigma_x^2} \quad (2.6)$$

Desvio cúbico médio (momento de terceira ordem)

O desvio cúbico, μ_3 , médio ou terceiro momento central dá uma medida da falta de simetria da distribuição, μ_3 . A partir deste momento obtém-se um coeficiente, adimensional, da assimetria, γ_1 .

$$\mu_3 = \int_{-\infty}^{+\infty} (x - \mu_x)^3 \times f(x) dx \quad (2.7)$$

$$\gamma_1 = \frac{\mu_3}{\sigma^3} \quad (2.8)$$

Desvio quártico médio (momento de quarta ordem)

O desvio quártico médio, μ_4 , é um parâmetro que permite verificar se a representação gráfica da distribuição tem uma forma mais ou menos achatada. Este desvio permite ainda calcular um coeficiente de achatamento adimensional, γ_2 também conhecido por curtose. Se este coeficiente for maior que zero, a distribuição apresenta uma maior concentração a volta da media, estando por isso pouco achatada.

$$\mu_4 = \int_{-\infty}^{+\infty} (x - \mu_x)^4 \times f(x) dx \quad (2.9)$$

$$\gamma_2 = \frac{\mu_4}{\sigma^4} - 3 \quad (2.10)$$

2.1.3. DISTRIBUIÇÕES DE PROBABILIDADE COMUNS EM ENGENHARIA CIVIL

Através dos parâmetros de distribuição consegue-se definir qualquer distribuição de probabilidade. As seguintes distribuições são as mais correntes na engenharia e são também as que vão estar presentes no programa automático de apoio à avaliação de fiabilidade estrutural.

2.1.3.1. Leis de probabilidade discretas

Lei binomial, $B(n,p)$

Muitos ensaios realizados na área da engenharia têm um carácter repetitivo e podem ser registados em termos de ocorrência e não ocorrência, resiste e não resiste, etc. Isto é, só dois resultados são possíveis. Por exemplo a contagem do número de vezes que um determinado acontecimento ocorrer ao longo de uma série de experiências (tijolos defeituosos numa paleta). Se uma experiência só tem dois resultados possíveis, os resultados associados a cada experiência são independentes e a probabilidade de ocorrência e não ocorrência mantém-se constante, pode-se considerar que é uma experiência de

Bernoulli e pode ser representada matematicamente por uma distribuição binomial. Assim a lei binomial determina a probabilidade de obter “x” sucessos em “n” tentativas.

Função densidade de probabilidade:

$$p(x) = C_x^n \times p^x \times (1-p)^{n-x}, \quad x = 0,1,2, \dots, n \quad (2.11)$$

Média:

$$\mu_X = n \cdot p \quad (2.12)$$

Desvio padrão:

$$\sigma_X = \sqrt{n \times p \times (1-p)} \quad (2.13)$$

Coefficiente de assimetria:

$$\gamma_1 = \frac{1 - 2 \times p}{[n \times p \times (1-p)]^{\frac{1}{2}}} \quad (2.14)$$

Coefficiente de achatamento:

$$\gamma_2 = \frac{1 - 6 \times p \times (1-p)}{n \times p \times (1-p)} \quad (2.15)$$

Notas:

- $C_x^n = \frac{n!}{x!(n-x)!}$
- p é a probabilidade de sucesso numa experiência de duas alternativas possíveis.

Lei geométrica, G(p)

Em engenharia é importante prever quando um acontecimento vai ocorrer pela primeira vez. Numa experiência de Bernoulli, com a probabilidade de ocorrer igual a p , a probabilidade de ocorrer à n -ésima experiência, depois da primeiras $n-1$ tentativas terem sido não ocorrência, é dada pela distribuição geométrica.

Função de distribuição:

$$F(n) = \sum_{k=0}^n (1-p)^{k-1} \times p = 1 - (1-p)^n, \quad n = 1,2,3 \dots \quad (2.16)$$

Função densidade de probabilidade:

$$p(n) = (1-p)^{n-1} \times p, \quad m = 1,2,3 \dots \quad (2.17)$$

Média:

$$\mu_N = 1/p \quad (2.18)$$

Desvio padrão:

$$\sigma_X = \sqrt{(1-p)/p^2} \quad (2.19)$$

Notas:

- p é a probabilidade de ocorrência numa experiência com duas alternativas possíveis.

Lei Binomial negativa, BN(k,p)

Numa sequência de experiências de Bernoulli, a variável aleatória t seguirá uma distribuição binomial negativa se representar o número de não ocorrências encontradas até suceder a k -ésima ocorrência.

Função densidade de Probabilidade:

$$p(t) = C_{k-1}^{t-1} (1-p)^{t-k} \times p^k, \quad t = k, k+1, \dots \quad (2.20)$$

Média:

$$\mu_T = k \times (1-p)/p \quad (2.21)$$

Desvio padrão:

$$\sigma_X = \sqrt{k \times (1-p)/p^2} \quad (2.22)$$

Coefficiente de assimetria:

$$\gamma_1 = \frac{2-p}{[k \times (1-p)]^{\frac{1}{2}}} \quad (2.23)$$

Coefficiente de achatamento:

$$\gamma_2 = \frac{6 \times (1-p) + p^2}{k \times (1-p)} \quad (2.24)$$

Notas:

- $C_{k-1}^{t-1} = \frac{(t-1)!}{(k-1)!(t-k)!}$
- p é a probabilidade de ocorrência numa experiência com duas alternativas possíveis.

Lei de Poisson, P(λ)

A distribuição de Poisson permite descrever um vasto conjunto de fenómenos aleatórios em que os acontecimentos se repetem no tempo ou no espaço. Ela expressa, por exemplo, a probabilidade de um certo número de eventos ocorrerem num dado período de tempo, caso estes ocorram com uma taxa média conhecida e caso cada evento seja independente do tempo decorrido desde o último evento.

Função de distribuição:

$$F(x) = \sum_{K=0}^x \frac{\lambda^K}{K!} \times e^{-\lambda} \quad (2.25)$$

Função densidade de Probabilidade:

$$p(x) = \frac{\lambda^k}{k!} \times e^{-\lambda} \quad (2.26)$$

Média:

$$\mu_X = \lambda \quad (2.27)$$

Desvio padrão:

$$\sigma_x = \sqrt{\lambda} \quad (2.28)$$

Coefficiente de assimetria:

$$\gamma_1 = \frac{1}{\sqrt{\lambda}} \quad (2.29)$$

Coefficiente de achatamento:

$$\gamma_2 = \frac{1}{\lambda} \quad (2.30)$$

2.1.3.2. Leis de probabilidade contínuas

Lei normal ou gausseana, $N(\mu, \sigma)$

A distribuição normal é uma das distribuições mais usadas em problemas de engenharia. Características como, poder ser aplicada para qualquer variável aleatória de $-\infty$ a $+\infty$, ser simétrica a volta da média, os valores da média, mediana e moda são semelhantes e ainda poder ser estimada directamente, tornam esta distribuição muito comum. Contudo, não é muito fácil efectuar o integral da função densidade de probabilidade. Resolve-se esta questão transformando a variável aleatória X numa variável com média igual a zero e desvio padrão igual a 1 (distribuição padronizada). Na figura 2.3 estão representadas curvas de quatro variáveis aleatórias com distribuição normal, com médias iguais mas desvios padrão diferentes.

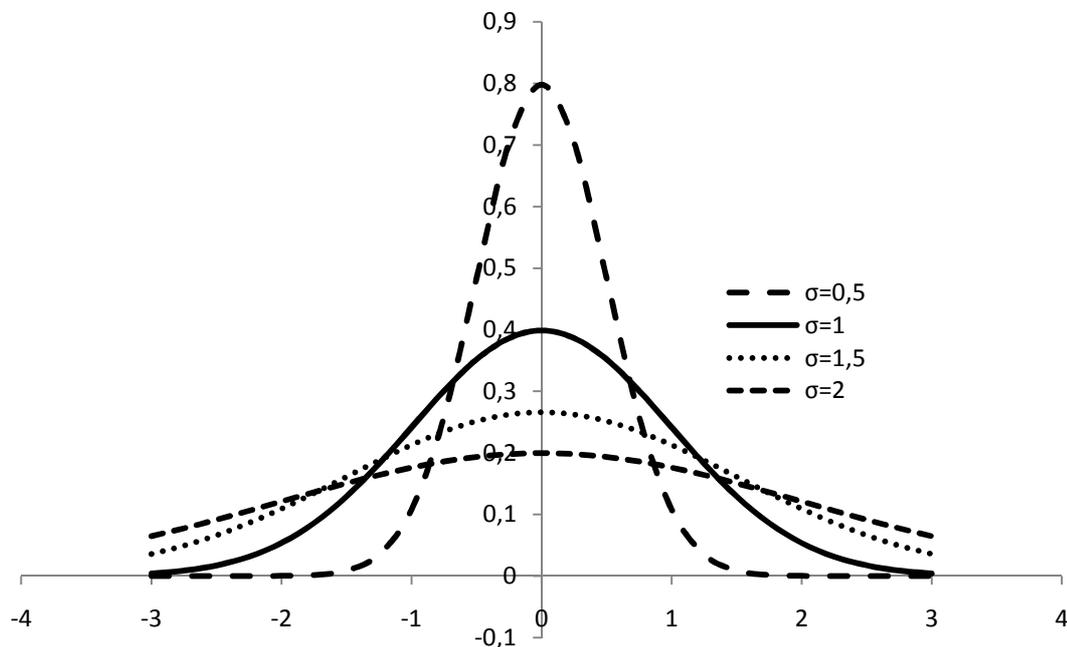


Fig. 2.3- Representação gráfica da função densidade de probabilidade gausseana.

Função de distribuição:

$$F(x) = \frac{1}{\sqrt{2\pi} \times \sigma_X} \times \int_{-\infty}^x \exp \left[-\frac{1}{2} \left(\frac{x - \mu_X}{\sigma_X} \right)^2 \right] dx, \quad x \in \mathfrak{R} \quad (2.31)$$

Função densidade de probabilidade:

$$f(x) = \frac{1}{\sqrt{2\pi} \times \sigma_X} \times \exp \left[-\frac{1}{2} \left(\frac{x - \mu_X}{\sigma_X} \right)^2 \right], \quad x \in \mathfrak{R} \quad (2.32)$$

Média:

$$\mu_X \quad (2.33)$$

Desvio padrão:

$$\sigma_X \quad (2.34)$$

Coefficiente de assimetria:

$$\gamma_1 = 0 \quad (2.35)$$

Coefficiente de achatamento:

$$\gamma_2 = 0 \quad (2.36)$$

Notas:

- A lei normal pode ser obtida da lei normal reduzida efectuando a seguinte transformação:
 $T = \frac{X - \mu_X}{\sigma_X}$, com $\mu_T = 0$ e $\sigma_T = 1$.

Lei lognormal, LN(λ, ξ)

Devido à natureza física de algumas variáveis aleatórias, valores negativos não são aceites. Nestes casos é mais apropriado usar a distribuição lognormal. Uma variável com esta distribuição tem o logaritmo natural com uma distribuição gausseana. Isto é, se Y é uma variável aleatória com distribuição gausseana, $X = \exp(Y)$ tem uma distribuição lognormal. Mais uma vez recorre-se à distribuição padronizada para estimar a probabilidade.

Função de distribuição:

$$F(x) = \frac{1}{\sqrt{2\pi} \times \xi} \times \int_{-\infty}^x \frac{1}{x} \times \exp \left[-\frac{1}{2} \left(\frac{\ln x - \lambda}{\xi} \right)^2 \right] dx, \quad x \in \mathfrak{R}^+ \quad (2.37)$$

Função densidade de probabilidade:

$$f(x) = \frac{1}{\sqrt{2\pi} \times \xi} \times \frac{1}{x} \times \exp \left[-\frac{1}{2} \left(\frac{\ln x - \lambda}{\xi} \right)^2 \right], \quad x \in \mathfrak{R}^+ \quad (2.38)$$

Média:

$$\mu_X = \exp \left(\lambda + \frac{1}{2} \xi^2 \right) \quad (2.39)$$

Desvio padrão:

$$\sigma_X = \mu_X \times \sqrt{e^{\xi^2} - 1} \quad (2.40)$$

Coefficiente de assimetria:

$$\gamma_1 = \left[\exp^2(\xi^2) + 2 \right] \times \left[\exp(\xi^2) - 1 \right]^2 \quad (2.41)$$

Coefficiente de achatamento:

$$\gamma_2 = \exp^4(\xi^2) + 2\exp^3(\xi^2) + 3\exp^2(\xi^2) - 6 \quad (2.42)$$

Notas:

- A lei lognormal pode ser obtida da lei normal reduzida efectuando a seguinte transformação: $T = \frac{\ln X - \lambda}{\xi}$.
- Os parâmetros λ e ξ correspondem à média e ao desvio padrão de $\ln X$, ou seja: $\lambda = \mu_{\ln X}$ e $\xi = \sigma_{\ln X}$

Lei beta, $\beta(a,b)$

A distribuição beta é conhecida pela sua versatilidade e é usada quando uma variável aleatória está delimitada por dois limites, a e b . Como a distribuição gausseana está limitada por $-\infty$ a $+\infty$ e a lognormal por 0 a $+\infty$, a distribuição beta está de a a b .

Função densidade de probabilidade:

$$f(x) = \frac{x^{a-1} \times (1-x)^{b-1}}{\beta(a,b)} \quad , \quad 0 \leq x \leq 1 \quad , \quad a > 0 \quad , \quad b > 0 \quad (2.43)$$

Sendo a função $\beta(a,b)$ dada por:

$$\beta(a,b) = \int_0^1 x^{a-1} \times (1-x)^{b-1} dx = \frac{\Gamma(a) \times \Gamma(b)}{\Gamma(a+b)} = \frac{(a-1)! (b-1)!}{(a+b-1)!} \quad (2.44)$$

Média:

$$\mu_X = \frac{a}{a+b} \quad (2.45)$$

Desvio padrão:

$$\sigma_X = \left[\frac{a \times b}{(a+b-2) \times (a \times b)^2} \right]^{1/2} \quad (2.46)$$

Coefficiente de assimetria:

$$\gamma_1 = \frac{2 \times (b-a) \times (a+b-1)^{1/2}}{(a+b+1) \times (a \times b)^{1/2}} \quad (2.47)$$

Coefficiente de achatamento:

$$\gamma_2 = \frac{3 \times (a + b) \times (a + b + 1) \times (a + 1) \times (2 * b - a)}{a \times b \times (a + b + 2) \times (a + b + 3)} + \frac{a \times (a - b)}{a - b} - 3 \quad (2.48)$$

Nota:

- $\Gamma(k)$ é a função gama e define-se por: $\Gamma(k, x) = \int_0^x e^{-u} u^{k-1} du$. Esta fórmula encontra-se tabelada em diversos livros.

Lei dos extremos tipo I ou de Gumbel, E-I(u,α) – Distribuição de máximos

Em muitas aplicações usadas em engenharia, é importante saber as distribuições dos valores máximos e mínimos das variáveis aleatórias. O tipo I desta distribuição é maioritariamente usado para definir a distribuição de valores máximos relativos a fenómenos naturais como vento e cheias. Esta distribuição pode resultar de amostras de distribuições gausseanas.

Função de distribuição:

$$F(x) = \exp[-e^{-\alpha(x-u)}], \quad \alpha > 0, \quad x \in \mathfrak{R} \quad (2.49)$$

Função densidade de probabilidade:

$$f(x) = \alpha \times \exp[-\alpha(x-u) - e^{-\alpha(x-u)}], \quad \alpha > 0, \quad x \in \mathfrak{R} \quad (2.50)$$

Média:

$$\mu_x = u + \gamma/\alpha \quad (2.51)$$

Desvio padrão:

$$\sigma_x = \frac{\pi}{\alpha\sqrt{6}} \quad (2.52)$$

Coefficiente de assimetria:

$$\gamma_1 = 1.1396 \quad (2.53)$$

Coefficiente de achatamento:

$$\gamma_2 = 2.4 \quad (2.54)$$

Nota:

- Constante de Euler: $\gamma = 0.5772156649 \dots$

Lei dos extremos tipo II ou de Frechet, E-II(u,k) – Distribuição de máximos

O tipo II desta distribuição é também maioritariamente usado para definir distribuições de valores máximos relativos a fenómenos naturais, mas que tenham uma cauda mais alongada, como a acção de sismos. Assim a distribuição de tipo II pode resultar de amostras de distribuições lognormais.

Função de distribuição:

$$F(x) = \exp\left[-\left(\frac{u}{X}\right)^k\right], \quad x \geq 0 \quad (2.55)$$

Função densidade de probabilidade:

$$f(x) = \frac{k}{X} \left(\frac{u}{X}\right)^k \exp\left[-\left(\frac{u}{X}\right)^k\right], \quad x \geq 0 \quad (2.56)$$

Média:

$$\mu_X = u \times \Gamma\left(1 - \frac{1}{k}\right), \quad k > 1 \quad (2.57)$$

Desvio padrão:

$$\sigma_X = u \times \left[\Gamma\left(1 - \frac{2}{k}\right) - \Gamma^2\left(1 - \frac{1}{k}\right)\right]^{\frac{1}{2}} \quad (2.58)$$

Nota:

- $\Gamma(k)$ é a função gama e define-se por: $\Gamma(k, x) = \int_0^x e^{-u} u^{k-1} du$. Esta fórmula encontra-se tabelada em diversos livros.

Lei dos extremos tipo III ou de Weibull, E-III(u,k) – Distribuição de mínimos

O tipo III desta distribuição é aplicado para descrever a distribuição os valores mínimos, sendo a distribuição dos valores máximos de interesse reduzido. É usado, geralmente, para determinar a resistência de materiais.

Função de distribuição:

$$F(x) = 1 - \exp\left[-\left(\frac{x - \varepsilon}{u - \varepsilon}\right)^k\right], \quad \varepsilon \geq 0, \quad k > 0, \quad u > \varepsilon \geq 0 \quad (2.59)$$

Função de densidade:

$$f(x) = \frac{k}{u - \varepsilon} \left(\frac{x - \varepsilon}{u - \varepsilon}\right)^{k-1} \exp\left[-\left(\frac{x - \varepsilon}{u - \varepsilon}\right)^k\right], \quad \varepsilon \geq 0, \quad k > 0, \quad u > \varepsilon \geq 0 \quad (2.60)$$

Média:

$$\mu_X = \varepsilon + (u - \varepsilon) \times \Gamma\left(1 + \frac{1}{k}\right) \quad (2.61)$$

Desvio padrão:

$$\sigma_X = (u - \varepsilon) \times \left[\Gamma\left(1 + \frac{2}{k}\right) - \Gamma^2\left(1 + \frac{1}{k}\right)\right]^{\frac{1}{2}} \quad (2.62)$$

Nota:

- $\Gamma(k)$ é a função gama e define-se por: $\Gamma(k, x) = \int_0^x e^{-u} u^{k-1} du$. Esta fórmula encontra-se tabelada em diversos livros.

2.2. CONCEITOS DE SEGURANÇA DE ESTRUTURAL

2.2.1. INTRODUÇÃO

Em engenharia, o dimensionamento de uma estrutura envolve criar elementos que satisfaçam vários critérios de performance, como segurança, durabilidade e manutenção. Isto é, uma estrutura deve ser concebida de forma a resistir a todas as acções que serão aplicadas, no momento da sua entrada em funcionamento e durante toda a vida da estrutura e ainda que a sua manutenção e reabilitação seja a mais económica possível. Mas, como já foi referido, uma estrutura depende de diversos factores, a maioria dos quais têm associados incertezas, que podem ser definidas estatisticamente. Assim a avaliação da fiabilidade estrutural tem um carácter probabilístico.

Um dos principais objectivos da avaliação da fiabilidade estrutural é atingir os níveis de fiabilidade necessários, com o consumo mínimo de recursos. Geralmente a segurança é atingida havendo um excesso de capacidade resistente. Com a evolução das metodologias aplicadas este excesso tem vindo a ser reduzido.

A construção de estruturas começou por ser efectuada com base na experiência dos técnicos, baseando-se em princípios básicos e sem qualquer regulamento. Frequentemente, as estruturas eram construídas com um excesso de capacidade resistente. Com a evolução das metodologias de dimensionamento, começaram-se a estabelecer regras de avaliação de segurança com base científica, usando coeficientes de segurança. De início estes coeficientes eram muito limitados, mas com o avanço das técnicas de avaliação da segurança, surgiram coeficientes mais adequados. Contudo, até os mais recentes coeficientes de segurança não proporcionam um aproveitamento ideal para novas soluções construtivas. Para simplificar pode-se adaptar um coeficiente, já utilizado numa solução semelhante, mas o aproveitamento estrutural nunca será ideal. Como a construção civil é uma área que está em constante desenvolvimento, é necessário desenvolver uma metodologia que avalie adequadamente a segurança estrutural de todas as situações novas que possam surgir. A avaliação baseada em metodologias probabilísticas representa abordagens racionais do problema, permitindo considerar as variabilidades dos parâmetros estruturais de novas situações. Com a introdução da avaliação probabilística surgiram novos conceitos que servem como medidas de fiabilidade estrutural, como a probabilidade de rotura e o índice de fiabilidade.

Embora as metodologias puramente probabilísticas sejam as mais rigorosas, as técnicas baseadas nos coeficientes de segurança continuam a ser mais usadas, desde que sejam calibrados por métodos probabilísticos de nível superior. Contudo vários conceitos probabilísticos têm vindo a ser introduzidos em vários códigos de dimensionamento.

Na presente secção, 2.2., serão abordados os conceitos associados à avaliação de segurança estrutural, como as incertezas associadas à avaliação de segurança de estruturas, estados limites, níveis de fiabilidade, etc. Será ainda feita uma comparação entre as abordagens à avaliação de segurança existentes.

2.2.2. INCERTEZAS ASSOCIADAS À AVALIAÇÃO DE SEGURANÇA DE ESTRUTURAS

Quando um engenheiro inicia um projecto, o seu objectivo principal é ser o mais preciso possível para garantir segurança absoluta. Contudo atingir a segurança absoluta é impraticável, devido às inúmeras incertezas associadas à construção civil, que não são passíveis de ser eliminadas. Tal deve-se ao desconhecimento das características, comportamento e parâmetros de um sistema estrutural. Com o aumento da complexidade dos sistemas estruturais, as incertezas tendem a aumentar em número

tornando-se mais difíceis de identificar. Estas incertezas provêm de inúmeras fontes, mas principalmente devido a quatro fontes:

- Incapacidade de prever as condições de carga a que a estrutura vai ser solicitada durante a sua vida útil;
- Incapacidade de prever as propriedades mecânicas dos materiais utilizados;
- Uso de modelos simplificados para o dimensionamento da estrutura;
- Intervenção do homem na concepção da estrutura.

Em situações extremas uma aglomeração de parâmetros incertos pode levar a desvios significativos da realidade. Para melhor compreender a influência de cada parâmetro de incerteza houve a necessidade de desenvolver métodos que permitissem sistematizar todos os parâmetros envolvidos nos sistemas estruturais. Começou-se por dividir estes parâmetros em dois grupos, grupo de origem natural, e grupo de origem na intervenção do homem. Com o recurso a metodologias probabilísticas surgiram conceitos que introduziram novos métodos de cálculo. Através do uso de valores característicos e de cálculo, o conceito de estado limite permitiu estabelecer metas mais precisas a atingir na avaliação da segurança, levando a uma melhor compreensão do comportamento real de um sistema estrutural.

As inúmeras incertezas que surgem num sistema estrutural têm vindo a ser objecto de estudo por vários autores, como por exemplo Thoft-Christensen em 1982, Melchers em 1987, Ayyub em 1997, Henriques em 1998, Nowak em 2000, Haldar & Mahadevan em 2000, Delgado em 2002 e Veiga em 2008.

De forma unânime as incertezas têm vindo a ser agrupadas em 4 grupos distintos:

- *Incerteza física*: é resultante da incapacidade de prever as propriedades mecânicas dos materiais e a variabilidade e simultaneidade de diferentes acções. Fenómenos naturais como vento, neve e acções sísmicas não podem ser controladas. Este tipo de incerteza pode ser estimado com o recurso a grandes bases de dados e a controlos de qualidade conveniente, mas será sempre difícil de prever devido ao seu carácter imprevisível.
- *Incerteza na modelação*: os sistemas de análise estrutural utilizados são representações aproximadas do comportamento real da estrutura e dos materiais. O dimensionamento de estruturas é geralmente efectuado com recurso a métodos de cálculo ou ferramentas computacionais que utilizam as características essenciais, de um sistema estrutural, para efectuar os cálculos numéricos. Durante o processo, alguns pormenores são ignorados, levando a um desfasamento entre o comportamento teórico e o real. Usando uma variável auxiliar que represente a relação entre o comportamento teórico e o real pode-se incluir este tipo de incerteza numa análise probabilística.
- *Incerteza estatística*: este tipo de incerteza está associado à definição de variáveis. Estas são definidas com recurso a ferramentas estatísticas que necessitam de informação prévia, informação essa que é escassa. Isto dificulta a criação de distribuições, para variáveis aleatórias, livres de incertezas. Para minimizar este tipo de incerteza é necessário um aumento de informação e redefinir a função de distribuição de forma a incorporar novos dados.
- *Incerteza devida a factores humanos*: este tipo de incerteza deve-se ao factor humano associado à construção civil. Estas incertezas não se devem unicamente à concepção da estrutura mas também à documentação, dimensionamento e utilização durante a vida útil. Pequenas acções humanas vão introduzindo erros no sistema estrutural, desde a omissão de dados até à má construção de um dado elemento. Este tipo de incerteza pode ser

atenuado recorrendo a equipas de projectistas e técnicos de construção experientes, mas nunca será totalmente eliminado.

2.2.3. ESTADOS LIMITES

Com a introdução dos métodos de avaliação estrutural, baseados em metodologias probabilísticas, surgiram um series de novos conceitos. Um dos mais importantes, é o conceito dos estados limites.

Quando um engenheiro inicia o dimensionamento de uma estrutura não deve ter como único objectivo garantir segurança para os utilizadores, mas também garantir que a estrutura terá um bom comportamento em serviço. Isto é, satisfazer os utentes proporcionando um correcto funcionamento da estrutura durante toda a vida útil da mesma e ainda permitir uma manutenção económica. Assim estabelecem-se dois limites básicos, um de segurança e outro de funcionamento.

Um estado limite consiste na definição de um conjunto de parâmetros que não devem ser excedidos. Consequentemente, numa situação de rotura entende-se que os parâmetros de um estado limite são excedidos. Segundo as definições presentes no regulamento em vigor (RSA,1983;EC1-1,1994), entende-se por estado limite um estado a partir do qual se considera que a estrutura fica prejudicada total ou parcialmente na sua capacidade para desempenhar as funções que lhe foram atribuídas.

Os cálculos usados para verificar se os estados limites são excedidos, são baseados em modelos teóricos ou experimentais, obtendo valores suficientemente precisos. Para estimar as acções que serão aplicadas ao longo da vida útil da estrutura utilizam-se combinações de acções.

Em suma, pode-se dividir os estados limites em dois grupos:

- *Estado limite último*: se este limite é excedido a estrutura em causa poderá entrar em rotura, havendo colapso total ou parcial da mesma. A segurança dos utentes e dos equipamentos deixa de ser assegurada e é aconselhável o abandono do edifício. Distinguem-se três tipos de rotura, por excesso de deformação numa dada secção devido às tensões actuantes, por instabilidade de elementos estruturais ou da estrutura como um todo e por perda de equilíbrio estático.
- *Estado limite de utilização*: este limite foi estabelecido para garantir um bom funcionamento da estrutura. Quando este é ultrapassado surgem danos que não põem em causa a segurança da estrutura, mas sim a funcionalidade, durabilidade e estética. Este estado limite está dividido em 3 subgrupos que estão associados a um tempo de ocorrência. O primeiro refere-se a uma duração muito curta, correspondendo a poucas horas na vida útil de uma estrutura. O segundo, curta, corresponde a cerca de 5% da vida de uma estrutura e o terceiro grupo, longo, corresponde a 50% da vida da mesma.

2.2.4. VERIFICAÇÃO DA SEGURANÇA AOS ESTADOS LIMITES

Conhecidas as origens de incertezas associadas à avaliação de segurança de estruturas e definidos os conjuntos de parâmetros que um sistema estrutural não deve exceder (estados limites) torna-se necessário proceder à verificação da segurança estrutural. Estes métodos devem ponderar as incertezas associadas aos parâmetros envolvidos no sistema estrutural de forma a assegurar uma margem de segurança. A verificação de segurança encontra-se dividida em 4 grupos, do 0 ao 3, com complexidade crescente (Henriques, 1998).

- *Nível 0*: estes são os métodos mais simples, empregando análises puramente deterministas. Ou seja, as variáveis têm valores estritamente determinísticos e as

incertezas são consideradas através de coeficientes de segurança globais, baseadas em experiências passadas.

- *Nível 1*: os métodos neste nível têm todos, um formato semi-probabilístico e servem de base para definir os critérios de muitos códigos de dimensionamento estrutural usados actualmente. A verificação efectua-se com condições determinísticas resultantes da introdução do conceito de estado limite. As variáveis envolvidas são representadas pelos seus valores nominais e característicos, afectados por coeficientes de segurança parciais. Estes valores são obtidos através de um estudo estatístico da distribuição de cada variável.
- *Nível 2*: refere-se a métodos com carácter probabilístico, baseados na definição de variáveis envolvidas na avaliação de segurança, através das quais é possível conhecer a sua tendência central (geralmente valores médios) e a sua dispersão. A sua maior limitação é a utilização de métodos simplificados para a determinação da probabilidade de rotura. Estes métodos utilizam como medida de segurança o índice de fiabilidade, que está directamente ligado à probabilidade de rotura.
- *Nível 3*: as análises referem-se a métodos puramente probabilísticos que utilizam a distribuição conjunta de todas as variáveis envolvidas nos sistemas estruturais. A probabilidade pode ser calculada analiticamente, embora seja só possível para casos muito simples, e pode ser calculada através de métodos de simulação. Todavia a aplicação em casos reais pode ser limitada devido à complexidade dos sistemas estruturais e ao excessivo esforço computacional necessário.

2.2.5. MEDIDORES DE FIABILIDADE ESTRUTURAL

2.2.5.1. Probabilidade de rotura

Simplificadamente pode-se considerar que a fiabilidade estrutural envolve apenas duas variáveis, a capacidade resistente, R , e a solicitação de cargas S . Estas são descritas pelas respectivas funções de densidade de probabilidade, f_R e f_S . Utilizando uma função de estado limite $g(\)$ para avaliar a probabilidade de rotura, obtém-se a seguinte expressão:

$$g(R, S) = R - S \quad (2.63)$$

Então a fronteira que separa o domínio de segurança e de rotura (estado limite), dá-se pela seguinte expressão:

$$g(R, S) = R - S = 0 \quad (2.64)$$

Assim a rotura ocorre quando as cargas solicitadas no sistema estrutural excedem a capacidade da resistente da estrutura. Sendo assim a probabilidade de rotura pode-se calcular integrando a função de densidade de probabilidade conjunta das variáveis R e S , $f_{R,S}$, no domínio da rotura.

$$p_f = P(g(R, S) \leq 0) \quad (2.65)$$

$$p_f = P(R - S \leq 0) \quad (2.66)$$

$$p_f = \iint_D f_{R,S}(R, S) dR dS \quad (2.67)$$

Admitindo a independência estatística das variáveis R e S, a função de densidade conjunta pode ser obtida de uma forma mais simples, substituindo a função de densidade de probabilidade conjunta pelo produto das respectivas funções de densidade de probabilidade marginais.

$$p_f = \iint_D f_R(r) * f_S(s) dr ds = \int_{-\infty}^{+\infty} \int_{-\infty}^{S>R} f_R(r) * f_S(s) dr ds \quad (2.68)$$

O domínio em que as cargas solicitadas superam a capacidade resistente do sistema estrutural é contínuo e contém todos os valores possíveis de S que verifica esta situação. As figuras 2.4 e 2.5 são representações gráficas da probabilidade de rotura de um sistema estrutural. Aqui estão representados a função de densidade de probabilidade de cada variável, bem como a função conjunta, e ainda a delimitação entre a zona de segurança e a de rotura.

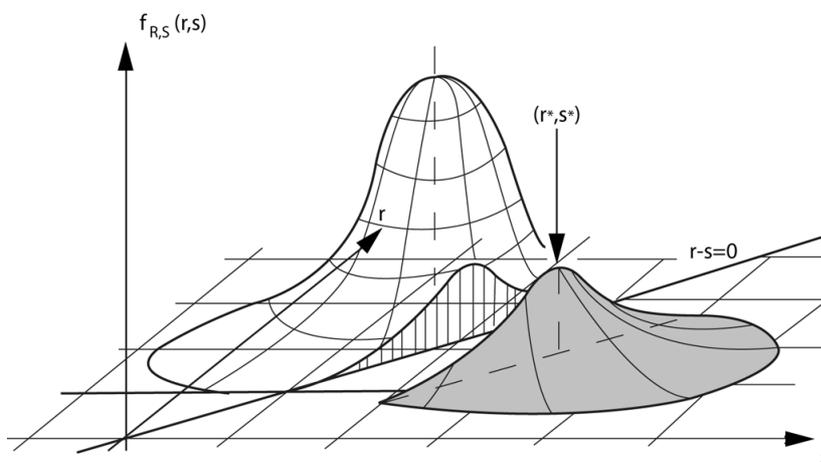


Fig. 2.4 - Representação gráfica da função de densidade de probabilidade conjunta Parte I (adaptado de Haldar & Mahadevan, 2000)

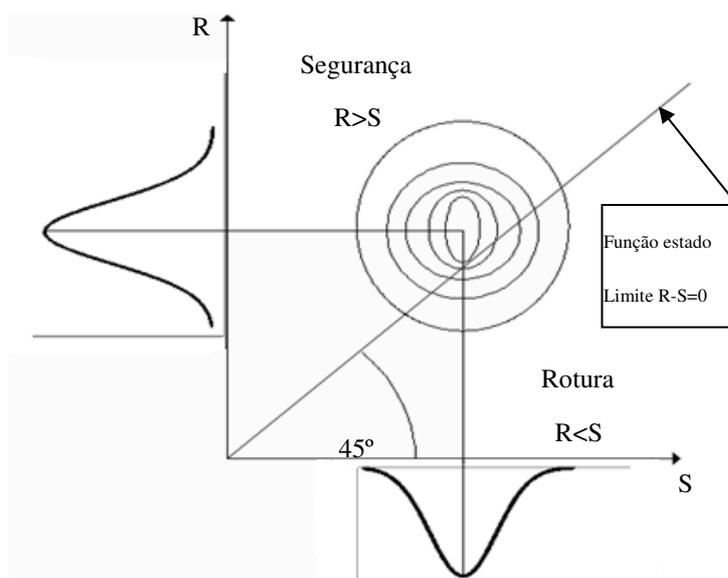


Fig. 2.5 - Representação gráfica da função de densidade de probabilidade conjunta Parte II (adaptado de Delgado, 2002)

Embora f_R e f_S sejam funções densidade de probabilidade das variáveis de R e S não existe razão para não serem funções da mesma variável, X. Assim a figura 2.4 dá lugar à figura 2.6 e a expressão (2.68) é modificada:

$$p_f = \int_{-\infty}^{+\infty} \int_{-\infty}^{S>R} f_R(x) \times f_S(x) dx \quad (2.69)$$

Observando o novo gráfico, da figura 2.6, consegue-se distinguir duas curvas, f_R e f_S , e ainda uma área sombreada. Esta área representa a probabilidade de as forças solicitadores serem maiores que a capacidade resistente do sistema estrutural. Estudando as características das duas distribuições de densidade é possível chegar à conclusão que esta área depende de dois factores: o primeiro factor é a posição relativa de cada curva, quanto maior for a distância relativa (média), menor será a probabilidade de rotura; o segundo factor refere-se à dispersão das curvas, se as curvas forem estreitas e desvio padrão for baixo, menor será a probabilidade de rotura. Então, com apenas dois dados é possível tirar conclusões rápidas.

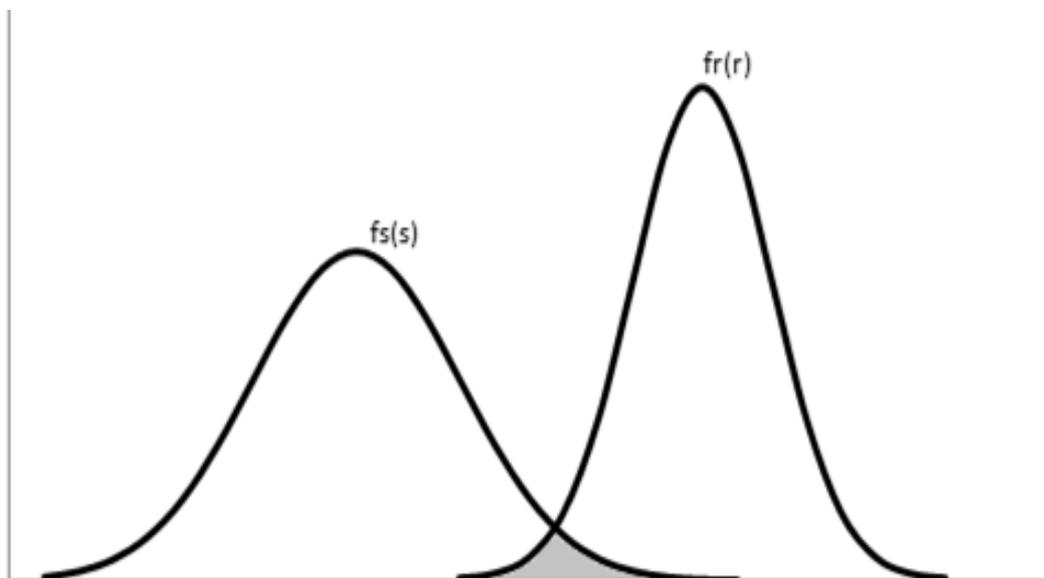


Fig. 2.6 – Representação gráfica das funções f_R e f_S , simplificada

Analisando os conceitos estabelecidos nos capítulos anteriores, nota-se que a função de distribuição pode ser obtida pela expressão 2.2. Então, sabendo como se calcula uma distribuição limitada na face superior, é fácil obter a função de distribuição resistente F_R .

$$p_f = \int_{-\infty}^{+\infty} F_R(x) \times f_S(x) dx \quad (2.70)$$

A expressão 2.70 representa a probabilidade de rotura e é representada por um integral, geralmente intitulado por integral de convolução. Este integral corresponde ao somatório das situações em que as cargas solicitadas são superiores à capacidade resistente do sistema estrutural.

2.2.5.2. Índice de fiabilidade

Visto que o integral de convolução é de difícil obtenção, surgiu a necessidade de desenvolver metodologias que permitissem a obtenção da probabilidade de rotura, directa ou indirectamente. Vários autores elaboraram metodologias de análise baseadas na teoria da fiabilidade, mas só em 1969 Cornell encontrou a primeira solução viável. Cornell apresentou o primeiro método de fiabilidade, intitulado por método *FOSM* (*First Order Second moment Reliability Method*). Neste método foi introduzido, pela primeira vez o conceito do índice de fiabilidade, β , a partir do qual é possível calcular a probabilidade de rotura por avaliação indirecta. Embora a introdução do índice de fiabilidade, também conhecido por índice de Cornell, tenha sido um grande avanço na área de fiabilidade estrutural, o método FOSM apresentou algumas limitações. A incapacidade de incluir informação sobre a distribuição das variáveis, suscitou o desenvolvimento de novas metodologias, culminando com o aparecimento das metodologias de segunda ordem na década de 80. Conhecidas por *SORM* (*Second Order Reliability Method*) permitiram ultrapassar a limitações proporcionadas pela metodologia *FOSM*.

Desde a sua introdução, por parte de Cornell, o índice de fiabilidade tem vindo a ser usado com mais frequência como uma medida de segurança, principalmente no âmbito da engenharia civil.

Admita-se que as variáveis R e S são, respectivamente, variáveis representativas da capacidade resistente de uma estrutura e a solicitação e ainda que estas são estatisticamente independentes com distribuições normais. Sendo a variável Z a condição fronteira:

$$Z = R - S \quad (2.71)$$

Então a expressão 2.66 é transformada na seguinte:

$$P_f = P(R - S \leq 0) = P(Z \leq 0) \quad (2.72)$$

Tendo em conta a propriedade aditiva destas variáveis, pode-se afirmar que a variável Z tem uma distribuição normal e os seus parâmetros de caracterização podem ser definidos da seguinte forma:

$$\mu_Z = \mu_R - \mu_S \quad (2.73)$$

$$\sigma_Z = \sqrt{\sigma_R^2 + \sigma_S^2} \quad (2.74)$$

Com as características da variável Z conhecidas procede-se ao cálculo da probabilidade de rotura. Recorrendo à função da lei normal reduzida, média nula e desvio padrão unitário, transforma-se a variável Z numa variável normal reduzida.

$$P_f = P(Z \leq 0) = \Phi\left(\frac{0 - \mu_Z}{\sigma_Z}\right) \quad (2.75)$$

Sabendo que o índice de fiabilidade é:

$$\beta = \frac{\mu_Z}{\sigma_Z} \quad (2.76)$$

a probabilidade de rotura passa a ser calculada pela seguinte expressão:

$$P_f = P(Z \leq 0) = \Phi\left(\frac{-(\mu_R - \mu_S)}{\sqrt{\sigma_R^2 + \sigma_S^2}}\right) = \Phi(-\beta) \quad (2.77)$$

Para melhor compreender a relação entre o índice de fiabilidade e a probabilidade de rotura vai-se analisar o gráfico na figura 2.7 e a tabela 2.1. Na figura 2.7 está representada uma função densidade de

probabilidade da variável Z . Como é possível observar quanto maior for o índice de fiabilidade maior será a margem de segurança e, por consequência, menor a probabilidade de rotura. Na tabela 2.1 podemos ver a relação numérica entre a probabilidade de rotura e o índice de fiabilidade.

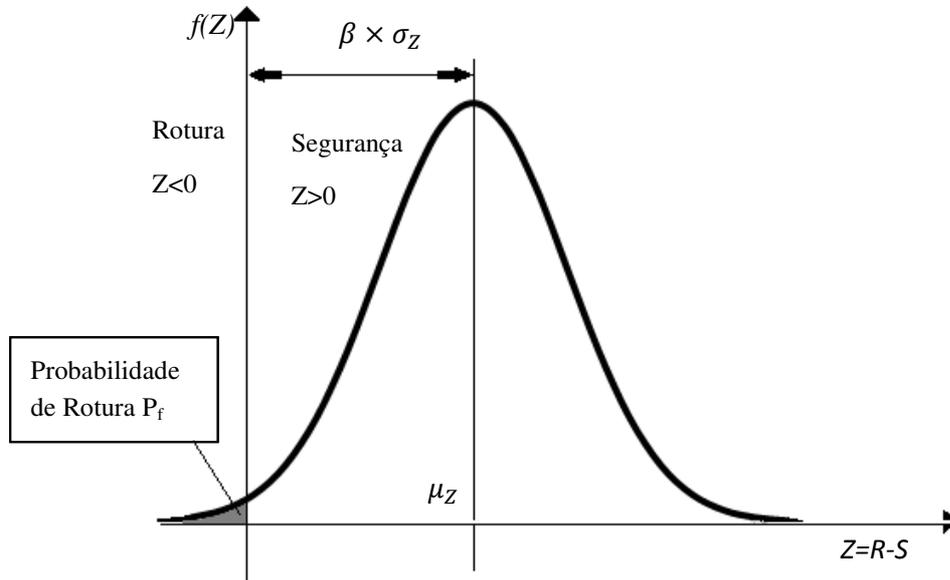


Fig. 2.7 - Representação gráfica da relação do índice de fiabilidade com a probabilidade de rotura (adaptado de Delgado, 2002)

Tabela 2.1 - Relação entre a probabilidade de rotura e o índice de fiabilidade

Probabilidade de Rotura p_f	Índice de fiabilidade β
$0.5 \cdot 10^0$	0.00
$1.0 \cdot 10^{-1}$	1.28
$1.0 \cdot 10^{-2}$	2.33
$1.0 \cdot 10^{-3}$	3.09
$1.0 \cdot 10^{-4}$	3.72
$1.0 \cdot 10^{-5}$	4.27
$1.0 \cdot 10^{-6}$	4.69
$1.0 \cdot 10^{-7}$	5.20
$1.0 \cdot 10^{-8}$	5.61
$1.0 \cdot 10^{-9}$	6.00

Com a intenção de estabelecer limites inferiores para índices de fiabilidade, foi publicado no EC-0 (2002), nos anexos B e C, valores mínimos recomendados. Aqui estabelecem-se valores mínimos para

o estado limite de último (ELU), para o estado limite de serviço (ELS) e para o estado limite de fadiga (ELF). Estes valores estão ainda divididos por dois conjuntos de grupos, por classes de fiabilidade (RC – Reliability class) e por anos de referência. As classes de fiabilidade referem-se ao grau das consequências de perda de vidas humanas, económicas, sociais e ambientais, sendo RC1 de consequência baixa, RC2 de consequência média e RC3 de consequência alta. Na tabela 2.2 apresenta-se uma síntese dos valores mínimos recomendados para o índice de fiabilidade.

Tabela 2.2 - Valores Mínimos recomendados para o índice de fiabilidade (EC-0, 2002).

	Período de referência de 1 ano			Período de referência de 50 anos		
	ELU	ELS	ELF	ELU	ELS	ELF
RC1	4.2	-	-	3.3	-	-
RC2	4.7	2.9	-	3.8	1.5	1.5 até 3.8*
RC3	5.2	-	-	4.3	-	-

* - Nota: este valor depende do grau de inspecção, reparação e tolerância ao dano.

Num caso em que as variáveis a analisar não têm distribuições normais, a probabilidade de rotura calculada não apresenta um valor exacto, sendo o índice de fiabilidade um valor indicativo. Todavia quanto maior for o índice menor será a probabilidade de rotura, para qualquer caso.

Até agora considerou-se que o índice de fiabilidade é dependente de duas variáveis elementares, capacidade resistente R e solicitação S . Num sistema estrutural tal não acontece, ambas as variáveis são compostas por um conjunto de variáveis, de forma que:

$$Z = G(X_1, X_2, \dots, X_m) = 0 \quad (2.78)$$

$$G(X) = a_0 + a_1X_1 + \dots + a_nX_n = a_0 + \sum_{i=1}^n a_iX_i \quad (2.79)$$

Se as variáveis constituintes de R e S forem estatisticamente independentes, com distribuição normal e a função de estado limite for linear, então, graças à propriedade aditiva das distribuições normais, a determinação do valor médio σ_Z e μ_Z é imediata através da expressão 2.80 que é semelhante à expressão (2.76).

$$\beta = \frac{a_0 + \sum_{i=1}^n a_i\mu_i}{\sqrt{\sum_{i=1}^n \sum_{j=1}^n (a_i a_j C_{ij})}} \quad (2.80)$$

$$\mu_Z = \mu_R - \mu_S \quad (2.81)$$

$$\sigma_Z = \sqrt{\sigma_R^2 + \sigma_S^2} \quad (2.82)$$

onde μ é o vector das médias e C é a matriz de covariâncias.

Porém na maioria dos casos, nem todas as variáveis aleatórias têm uma distribuição normal e uma função de estado limite linear. Assim as propriedades aditivas deixam de ter efeito e o índice de fiabilidade não se calcula pelas expressões anteriores. A probabilidade terá de ser obtida através de uma generalização, aplicando uma integração múltipla de todas as variáveis aleatórias envolvidas:

$$p_f = \int \dots \int_{G(X) \leq 0} f_x(X_1, X_2, \dots, X_n) dx_1 dx_2 dx_n \quad (2.83)$$

Esta integração da função de densidade da probabilidade conjunta das variáveis aleatórias é praticamente impossível de efectuar. Assim recorre-se a simplificações ou métodos numéricos para resolver este integral. Existem dois tipos de abordagens para resolver o integral. O primeiro tipo, são os métodos de simulação estocástica que obtém resultados a partir de aproximações numéricas, como por exemplo o método de Monte Carlo e a sua variante método do Hipercubo Latino. O segundo tipo utiliza os métodos atrás referidos, *FORM* e *SORM*. Estes métodos baseiam-se, em aproximações lineares e quadráticas ao ponto mais provável de rotura na superfície de estado limite.

2.2.6. COMPARAÇÃO ENTRE OS MÉTODOS DE FIABILIDADE E OS MÉTODOS DE SIMULAÇÃO

Para proceder à análise da fiabilidade estrutural pode aplicar-se de métodos de fiabilidade e de simulação. Entre os métodos de fiabilidade, temos os *FORM/SORM*, baseados em cálculos analíticos com rigor geralmente suficiente. Entre os métodos de simulação, temos o método de Monte Carlo e as suas variantes, baseados no cálculo numérico de integrais com resolução analítica impraticável.

Embora os métodos de fiabilidade *FORM/SORM* não sejam objecto de estudo desta dissertação, será feita uma breve referência às suas potencialidades e limitações, culminando numa comparação qualitativa com os métodos de simulação mais usados. Informação mais detalhada, sobre os métodos de fiabilidade, pode ser encontrada na seguinte bibliografia: Haldar & Mahadevan (2000) e Henriques (1998).

As técnicas *FORM/SORM* aplicam-se a variáveis aleatórias contínuas e a sistemas em que a superfície de rotura seja também contínua com variação suave e gausseana. O seu rigor é aceitável se as condições anteriores forem respeitadas, sendo para probabilidades de ordem de grandeza muito pequena mais eficientes que os métodos de simulação. A grande dificuldade nestes métodos é a definição da superfície de rotura, aumentando o tempo de computação com o aumento de complexidade. Para casos em que a superfície de rotura apresenta uma complexidade demasiado elevada, tem-se um tempo de computação também elevado e o rigor é posto em causa.

A grande potencialidade dos métodos de simulação, é a possibilidade de serem aplicados a qualquer caso, qualquer tipo de variável aleatória e qualquer tipo de superfície de rotura. O erro associado a estes métodos pode ser controlado, aumentando o número de simulações. Quando este número tende para infinito os resultados tendem para um valor exacto. Todavia este rigor vem com uma contrapartida, o tempo de computação. Para reduzir o tempo de computação foram desenvolvidas técnicas de redução de variância. Este tema será desenvolvido mais à frente na secção 2.3.

Comparando agora os dois grupos de metodologias, pode-se concluir que, os métodos de fiabilidade requerem menos tempo de computação, mas são de aplicação limitada e os métodos de simulação requerem tempos de computação elevados mas são de aplicação geral. As estruturas actuais são cada vez mais complexas, com sistemas estruturais com comportamento não linear, conduzindo ao crescente uso dos métodos de simulação. Com o avanço na área computacional, começam-se a desenvolver máquinas suficientemente potentes para simular e obter resultados adequados em tempos aceitáveis.

2.3. MÉTODOS DE SIMULAÇÃO

2.3.1. INTRODUÇÃO

Com uma técnica de simulação simples, é possível calcular a probabilidade de rotura para funções de estados limites explícitas e implícitas, sem conhecer as técnicas analíticas e só com conhecimentos básicos em probabilidades e estatística (Haldar & Mahadevan, 2000).

Como já foi referido no capítulo anterior as técnicas de fiabilidade estrutural são de difícil aplicação devido às inúmeras variáveis aleatórias envolvidas e à complexidade dos sistemas estruturais modernos. Em modelos numéricos de análise de fiabilidade estrutural, baseados em elementos finitos, com comportamento não linear, a definição de uma função de estado limite é, por vezes, impraticável. Segundo Henriques (1998) existem 3 grupos de metodologias para contornar estas dificuldades: métodos de perturbação, métodos de fiabilidade dos quais se destaca o método de superfície de resposta e os métodos de simulação. Embora os dois primeiros grupos não sejam objecto de estudo desta dissertação será realizada uma breve referência para destacar as diferenças entre as mesmas e os métodos de simulação.

Nos métodos de perturbação começa-se por caracterizar as variáveis aleatórias envolvidas, médias, desvios padrão e respectivas distribuições probabilísticas. De seguida, de forma determinística, calcula-se o valor médio e o desvio padrão da resposta do sistema estrutural. Ao introduzir perturbações nas variáveis envolvidas cálculo da resposta média, obtém-se a matriz de covariâncias. A avaliação da fiabilidade estrutural faz-se através da matriz de covariâncias e da resposta média.

Dos métodos de fiabilidade destaca-se o método de superfície de resposta. Como já foi referido anteriormente, numa análise de fiabilidade por métodos de fiabilidade, a maior dificuldade reside na definição da função de estado limite. O método de superfície de resposta consiste em substituir esta função de estado limite por uma função analítica simplificada. Esta metodologia torna possível a avaliação da probabilidade de rotura com um erro aceitável.

Na forma mais simples os métodos de simulação pretendem amostrar várias vezes, todas as variáveis envolvidas num sistema estrutural, para representar a sua distribuição probabilística. De seguida calcular o problema deterministicamente, para cada combinação de variáveis aleatórias. Repetindo este cálculo várias vezes, obtém-se a probabilidade geral de falha do sistema. Quando o número de ciclos tende para infinito o erro associado a este método tende para zero. Com o desenvolvimento computacional recente, tornou-se possível incluir poderosos algoritmos de análise estrutural, baseados em elementos finitos, que permitem calcular sistemas estruturais com comportamentos não lineares. Dos métodos de simulação destaca-se o método de Monte Carlo. Este é considerado o método de simulação pura de simples aplicação e que requer um número de simulações elevado para obter um erro aceitável. Para resolver este problema de excessivo esforço computacional, desenvolveram-se técnicas de redução de variância, dos quais resultou o método do Hipercubo Latino.

2.3.2. MÉTODO DE MONTE CARLO

Utilizando conhecimentos básicos de estatística e probabilidade, o método de Monte Carlo permite calcular a probabilidade de rotura ou verificar se uma função de estado limite é respeitada, de forma relativamente simples. Assim o método de Monte Carlo gera um grande conjunto de variáveis aleatórias que são processadas e que simulam uma situação real com precisão excelente. De forma geral este método consiste na simulação de um conjunto de variáveis aleatórias, tendo em conta as

suas distribuições probabilísticas, e na verificação da fiabilidade estrutural para cada conjunto por métodos determinísticos. Se o algoritmo for conhecido, é ainda possível incorporar comportamentos não lineares dos materiais e da estrutura, sendo assim uma ferramenta muito útil na engenharia civil. Na sua forma mais básica considera-se que este método é uma técnica de simulação pura.

O método de Monte Carlo pode-se dividir em 5 fases (figura 2.8):

- Definição dos parâmetros envolvidos: consiste em definir o sistema estrutural em termos de variáveis aleatórias, respectivas distribuições probabilísticas e parâmetros de distribuição.
- Simulação das variáveis: gerar n conjuntos de amostras aleatórias, de acordo com as respectivas distribuições.
- Análise estrutural: avaliar cada conjunto de amostras deterministicamente e contabilizar o número de ocorrências de rotura.
- Verificação da precisão: determinar a precisão da simulação, através do cálculo de erros. Se estes forem superiores a valores admissíveis, deve-se repetir o processo de simulação com um número de amostras, n , maior.
- Avaliação de segurança: Cálculo do índice de fiabilidade, da probabilidade de rotura e dos parâmetros estatísticos que traduzem o comportamento do sistema estrutural.

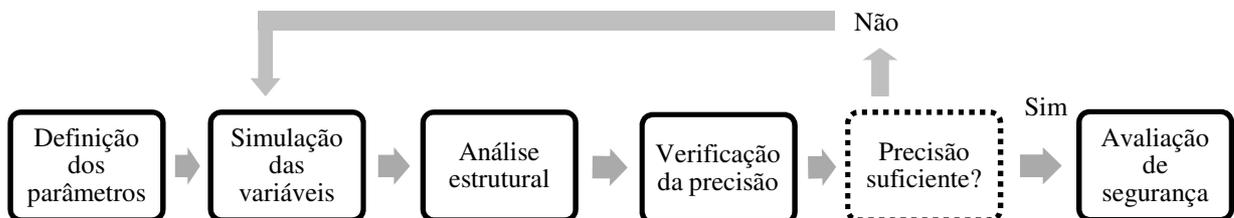


Fig. 2.8 - Fases do método de Monte Carlo

2.3.2.1. Geração de valores aleatórios

A base deste processo consiste em gerar números aleatórios uniformemente distribuídos no intervalo $]0, 1[$ aos quais, em função das características das funções de distribuição, se associam valores representativos das variáveis (Delgado, 2002).

A geração de valores aleatórios no método de Monte Carlo é muito simples e de fácil aplicação. Com todas as variáveis aleatórias do sistema estrutural definidas, caracterizadas pelas respectivas

distribuições e parâmetros de distribuição, procede-se a transformação inversa da lei teórica da distribuição acumulada, em causa. Sendo u_i uma variável uniformemente distribuída entre]0, 1[, intitulada por pseudo-aleatória, $F_X(x_i)$ a função de distribuição acumulada da variável aleatória X e $F_X^{-1}(u_i)$ a função inversa da distribuição acumulada, os valores aleatórios obtêm-se pela seguinte expressão:

$$F_X(x_i) = u_i \leftrightarrow F_X^{-1}(u_i) = X_i, \quad i = 1, 2, \dots, n. \quad (2.84)$$

Qualquer programa, com fins matemáticos, ou calculadora, tem uma função que permite calcular estes pseudo-aleatórios, sendo os seus valores raramente repetidos. Os algoritmos mais utilizados para gerar estes valores são dependentes de um parâmetro temporal, que muda sempre que a função é chamada. Este algoritmo é caracterizado por uma semente ou “seed”, que define o ponto de partida para a geração de valores aleatórios. Sabendo o número de amostras, n , que são necessárias, repete-se este processo aleatório até que hajam n valores aleatórios. Na figura 2.9 está representada a metodologia da transformação inversa que será repetida n vezes.

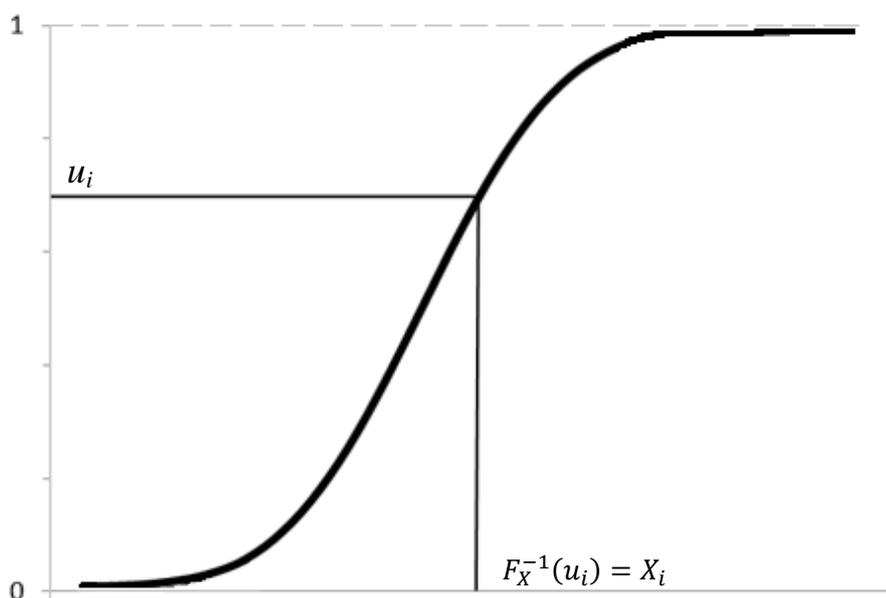


Fig. 2.9 - Representação gráfica da transformação inversa (adaptado de Veiga, 2008)

Para leis probabilísticas que não têm uma função de distribuição explícita, como a lei normal, não é possível efectuar o cálculo da função inversa da distribuição cumulativa, analiticamente. Para este tipo de distribuição é mais eficiente recorrer ao método de *Box-Müller*. Este método consiste na transformação de duas variáveis, U_1 e U_2 , aleatórias independentes contínuas e uniformes em]0, 1[, em duas variáveis, X_1 e X_2 , aleatórias independentes reduzidas, média nula e desvio padrão unitário (Henriques, 1998):

$$X_1 = (-2 \times \ln U_1)^{1/2} \times \cos 2\pi U_2 \quad (2.85)$$

$$X_2 = (-2 \times \ln U_1)^{1/2} \times \sin 2\pi U_2 \quad (2.86)$$

Uma distribuição de variáveis aleatórias independentes log-normalmente distribuídas pode ser obtida das expressões (2.85) e (2.86), fazendo a seguinte transformação: $Y_i = \ln X_i$.

Contudo o “software” de programação utilizado para elaborar o programa de apoio à avaliação de fiabilidade estrutural já tem uma função que efectua, automaticamente, as transformações necessárias.

Tendo as amostras disponíveis procede-se à avaliação da fiabilidade estrutural, efectuando um cálculo determinístico para cada amostra simulada e verificando se as funções de estado limite são cumpridas. Graças à simplicidade dos cálculos determinísticos aliados a este método de simulação, é possível incorporar qualquer tipo de comportamento estrutural, com especial destaque aos comportamentos não lineares. A adaptabilidade que esta metodologia proporciona, levou a que esta seja cada vez mais utilizada como ferramenta de avaliação de fiabilidade estrutural.

2.3.2.2. Teste de Hipóteses

Os testes de hipótese servem para avaliar a precisão da simulação estocástica do gerador de variáveis pseudo-aleatórias, estando divididos em dois grupos: gráficos e numéricos. Os testes de hipóteses gráficos consistem na análise das distribuições simuladas, através de diagramas como histogramas e distribuições acumuladas, como se vê nos gráficos da figura 2.10. Estes testes permitem uma rápida avaliação do sistema, sendo puramente qualitativos. Ou seja quando é necessário uma análise mais pormenorizada devem ser complementados com outro tipo de testes.

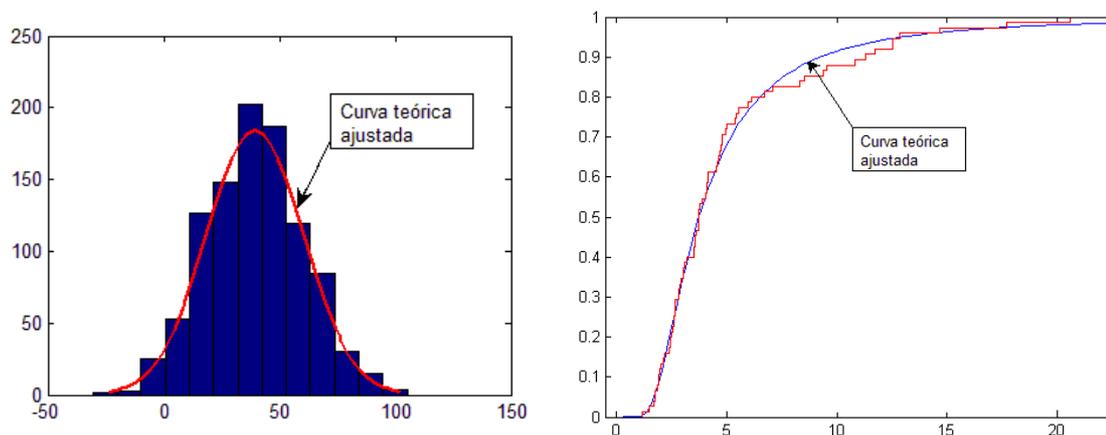


Fig. 2.10 - Histograma com curva teórica ajusta (à esquerda); curva de frequências acumuladas com curva teórica ajustada (à direita).

Entre os muitos tipos de testes numéricos destacam-se, o teste de normalidade, o teste de Kolmogorov-Smirnov e a sua variante, teste de Lilliefors. O teste de normalidade é aplicável somente a distribuições gausseanas e é baseada na estimativa dos coeficientes de achatamento e assimetria. No programa de apoio à avaliação de fiabilidade estrutural será utilizada a versão original do teste de Kolmogorov-Smirnov e a versão modificada. A diferença entres estes dois testes está na determinação dos parâmetros da lei probabilística. No teste original estes parâmetros são determinados a partir de valores fixados pelo utilizador (módulo um) e no teste modificado, são determinados a partir da amostra analisada (módulo dois). Assim esta diferença actua no cálculo do coeficiente de distribuição, que para o teste modificado é mais restritivo.

Teste de normalidade

Através dos coeficientes de assimetria γ_1 e de achatamento γ_2 , é possível verificar aproximadamente as características gausseanas de uma distribuição estatística de uma variável aleatória. Teoricamente uma

curva gausseana tem estes dois coeficientes nulos, mas tal não acontece. As inevitáveis flutuações da amostragem, introduzem o conceito de valores estimados dos coeficientes de assimetria, $\hat{\gamma}_1(n)$, e achatamento, $\hat{\gamma}_2(n)$. Então, consideram-se estatisticamente não significativos, todos os desvios em relação a zero dos valores $\hat{\gamma}_1(n)$ e $\hat{\gamma}_2(n)$, que não sejam superiores a uma vez e meio ou duas vezes dos desvios padrão das características correspondentes ($\sigma_{\hat{\gamma}_1}$ e $\sigma_{\hat{\gamma}_2}$).

- Estimativa dos coeficientes de assimetria:

$$\hat{\gamma}_1(n) = \frac{\frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^3}{\left[\frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^2 \right]^{3/2}}. \quad (2.87)$$

- Estimativa do coeficiente de achatamento:

$$\hat{\gamma}_2(n) = \frac{\frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^4}{\left[\frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^2 \right]^2} - 3. \quad (2.88)$$

- Os desvios padrão de $\hat{\gamma}_1(n)$ e $\hat{\gamma}_2(n)$ calculam-se por:

$$\sigma_{\hat{\gamma}_1} = \sqrt{\frac{6(n-2)}{(n+1)(n+3)}}; \quad (2.89)$$

$$\sigma_{\hat{\gamma}_2} = \sqrt{\frac{24n(n-2)(n-3)}{(n+1)^2(n+3)(n+5)}}. \quad (2.90)$$

- Se estas duas verificações forem cumpridas, aceitam-se os desvios como estatisticamente não significativos:

$$|\hat{\gamma}_1(n)| < 1.5 \times \sigma_{\hat{\gamma}_1}; \quad (2.91)$$

$$\left| \hat{\gamma}_2(n) + \frac{6}{n+1} \right| < 1.5 \times \sigma_{\hat{\gamma}_2}. \quad (2.92)$$

- Se uma das duas condições não verificar, procede-se à verificação do teste de Kolmogorov-Smirnov.
- A hipótese de normalidade é rejeitada, sem possível verificação do teste de Kolmogorov-Smirnov, se uma das seguintes condições não se verificar.

$$|\hat{\gamma}_1(n)| < 2 \times \sigma_{\hat{\gamma}_1} \quad (2.93)$$

$$\left| \hat{\gamma}_2(n) + \frac{6}{n+1} \right| < 2 \times \sigma_{\hat{\gamma}_2} \quad (2.94)$$

Teste de Kolmogorov-Smirnov (K-S) e Teste modificado por Lilliefors (1967)

Este teste compara a distribuição estatística da variável aleatória X com uma lei teórica de distribuição de qualquer tipo, quantificando a distância entre as duas. Ou seja, baseia-se na diferença das duas funções, $F(x) - F^*(x)$, sendo $F^*(x)$ a distribuição estatística (frequências acumuladas) da variável aleatória X e $F(x)$ a função de distribuição da lei teórica.

O processo de comparação entre as duas funções segue os seguintes passos (Henriques, 1998):

- Ordenação (de forma crescente) dos n valores observados (x_1, x_2, \dots, x_n);
- Cálculo das frequências acumuladas, $F^*(x)$, e da função distribuição da lei teórica considerada, $F(x)$, para todos os n valores;
- Cálculo do valor absoluto da máxima diferença:

$$D_n = \max_{1 \leq i \leq n} |F(x_i) - F^*(x_i)|. \tag{2.95}$$

- Cálculo do coeficiente de distribuição de \tilde{D}_n pelo método original e pelo método modificado, respectivamente:

$$\tilde{D}_n = D_n \times \sqrt{n} \quad , \tag{2.96}$$

$$\tilde{D}_n = D_n (\sqrt{n} + 0.12 + 0.11/\sqrt{n}). \tag{2.97}$$

- Verificação do ajuste através do cálculo da probabilidade (ver valores na tabela 3):

$$P(\tilde{D}_n < \lambda) = 1 - 2 \sum_{k=1}^{\infty} (-1)^k e^{-2k^2 \lambda^2}. \tag{2.98}$$

Se $P(\tilde{D}_n < \lambda) \leq \alpha$ rejeita-se a hipótese da lei teórica proposta, sendo α o nível significância exigido. No teste de Kolmogorov-Smirnov, os valores mais comuns para α são na ordem de 0,05.

Tabela 2.3 - Probabilidade do afastamento máximo entre $F^*(x)$ e $F(x)$ devido aos factores aleatórios seja não inferior ao valor observado.

\tilde{D}_n	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$P(\tilde{D}_n < \lambda)$	1.000	1.000	1.000	1.000	0.997	0.964	0.864	0.711	0.544	0.393	0.270
\tilde{D}_n	1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0	-
$P(\tilde{D}_n < \lambda)$	0.178	0.112	0.068	0.040	0.022	0.012	0.006	0.003	0.002	0.001	-

2.3.2.3. Cálculo do Coeficiente de variação e do erro

Existem dois métodos básicos para calcular a precisão da probabilidade de rotura, cálculo da variância ou coeficiente de variação (COV), contando o número de amostras que falham, N_f , entre as amostras simuladas, e cálculo do erro, ϵ , associado ao número de simulações realizadas. O cálculo da probabilidade de rotura pela divisão do número de falhas pelo número de simulações realizadas, é um método simplificado que pode conter erros consideráveis, especialmente para ciclos de simulações pequenos. Então é necessário desenvolver métodos que verifiquem a precisão deste método e do método indirecto, apresentado na secção 2.2.

O primeiro método consiste em considerar que qualquer simulação tem as características de uma experiência de Bernoulli, ou seja, considera-se que o número de falhas em n simulações tem uma distribuição binomial. Então a COV, é definida pela seguinte expressão (Haldrar & Mahadevan, 2000):

$$COV_{(P_f)} = \frac{\sqrt{\frac{(1 - P_f)P_f}{N}}}{P_f}, \quad (2.99)$$

onde $P_f = N_f/N$.

O segundo método pretende calcular o erro associado ao número de simulações efectuadas com um nível de confiança de 95%. A expressão (2.100) apresentada por Shooman, em 1968, indica que haverá cerca de 20% de erro se a probabilidade de rotura real P_f^R for 0,01 e se o número de simulações for 10000. Graças ao nível de confiança, 95%, pode-se garantir que a probabilidade de rotura não vai variar para além de 0.01 ± 0.002 . Analisando os dois métodos pode-se concluir que com o aumento de n simulações, o erro tende para zero e que quanto menor for a probabilidade de rotura mais simulações serão necessárias para obter um erro admissível.

$$\varepsilon\% = \sqrt{\frac{(1 - P_f^R)}{N \times P_f^R}} \times 200\%. \quad (2.100)$$

2.3.3. TÉCNICAS DE REDUÇÃO DE VARIÂNCIA – MÉTODO DO HIPERCUBO LATINO

As técnicas de redução de variância aumentam a eficiência e a precisão da avaliação de fiabilidade estrutural, usando relativamente poucas simulações; contudo, estas aumentam as dificuldades computacionais para cada simulação, sendo necessária uma quantidade considerável de conhecimentos para as implementar (Haldar & Mahadevan, 2000).

Para estimar a probabilidade de rotura com eficiência e precisão admissível, o método de Monte Carlo, necessita de um grande número de simulações. Sendo a probabilidade de rotura relativamente pequena na maioria dos sistemas estruturais, significa que a quantidade de simulações necessárias será demasiado grande, ou seja, demorado. Com os avanços modernos na área computacional um número elevado de simulações pode não ser um problema, mas com o aumento de complexidade dos sistemas estruturais, aumentam também a complexidade dos cálculos necessários para avaliar a fiabilidade estrutural. Surgiu a necessidade de reduzir o número de simulações, aumentando a eficiência das mesmas e mantendo a precisão. Criaram-se então técnicas de redução de variância para cumprir estes requisitos. Estas técnicas baseiam-se nos conhecimentos prévios sobre os sistemas para obter resultados com um número reduzido de simulações. As diferentes técnicas de redução de variância podem ser aplicadas em casos diferentes, dependendo da natureza do sistema em causa. Entre as várias técnicas de redução estão; a amostragem por importância, a amostragem estratificada e ainda o método do Hipercubo Latino, que pertence ao grupo das estratificadas.

A ideia base da amostragem por importância, é concentrar as amostras em pontos de maior interesse, que melhor contribuam para a probabilidade de rotura, em vez de considerar todo o espaço amostral. A amostragem estratificada consiste em dividir a distribuição da variável aleatória em diversas áreas, de forma a aplicar mais simulações nas regiões de maior importância, à semelhança da amostragem por importância. De seguida será descrito a geração aleatória de valores aleatórios para o caso do Hipercubo Latino.

2.3.3.1. Geração de valores aleatórios

A grande diferença entre o método de Monte Carlo e o método do Hipercubo Latino está no mecanismo gerador de amostras. Enquanto o primeiro utiliza um método relativamente simples, abordado na secção 2.3.2.1., o segundo já envolve um tratamento mais elaborado. Assim ao aplicar o método do Hipercubo Latino perde-se a simplicidade que estava associada ao método de Monte Carlo, mas atinge-se um nível de eficiência superior. O método do Hipercubo Latino encontra-se dividido em duas fases: na primeira gera-se um campo de valores possíveis para cada variável aleatória de acordo com respectivas distribuições; na segunda gera-se aleatoriamente n conjuntos de amostras.

Estando todas as variáveis aleatórias do sistema estrutural definidas, caracterizadas pelas respectivas distribuições e parâmetros de distribuição, começa-se por gerar um campo de valores possíveis para cada variável. Ou seja, sabendo o número de amostras, n , que são desejadas, prepara-se um conjunto de valores para cada variável aleatória. Para gerar este campo de valores é necessário dividir a função de densidade de probabilidade em n fatias de áreas iguais e obter a posição do centro de massa de cada uma. A determinação destas áreas e dos seus centros de massa envolvem cálculos complexos, sendo mais simples recorrer à função inversa da distribuição acumulada. Se em vez de usar a função de densidade de probabilidade, usar-se a função inversa da distribuição acumulada, $F_X(x_i)$ expressão (2.101), é possível calcular a posição do centro de massa de cada fatia, recorrendo a uma divisão em $2*n$ partes iguais do eixo y , ou seja da probabilidade (figura 2.11).

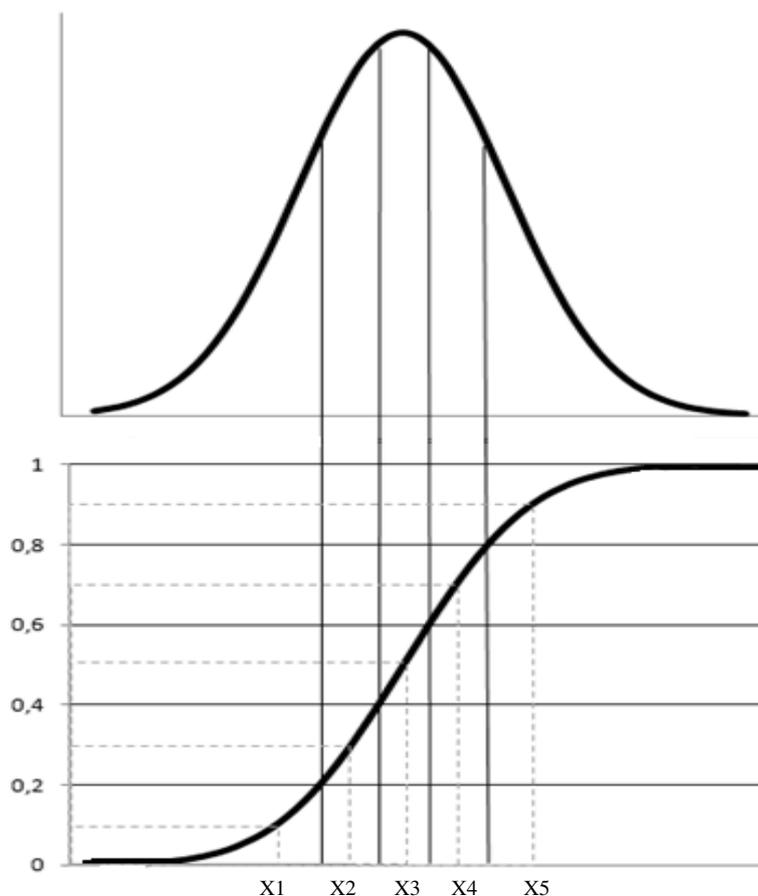


Fig. 2.11 - Representação gráfica da primeira fase do método do Hipercubo Latino

$$F_X(x_i) = X_i \leftrightarrow X_1 = F_X^{-1}(x_i) , \quad i = 1, 2, \dots, n . \quad (2.101)$$

Com os campos de valores para cada variável aleatória gerados, procede-se à geração de amostras aleatórias (segunda fase). Supondo que os valores de uma variável aleatória com n valores possíveis, estão armazenados num vector $V (X_1, X_2, \dots, X_n)$ e que a letra i corresponde à posição de cada valor no vector, o valor aleatório, v , para qualquer amostra, obtém-se pelas seguintes expressões:

$$i = \text{int} [\text{rand}(n)] + 1 \quad (2.102)$$

$$v = V(i) . \quad (2.103)$$

onde: $\text{rand}(n)$ gera um número aleatório com distribuição uniforme entre $]0,1[$ e multiplica-o por n ; $\text{int} []$ devolve a parte inteira do número.

Outra característica que destaca o método do Hipercubo Latino do método de Monte Carlo, são os valores gerados na primeira fase deste método (armazenados no vector V), estes não se podem repetir. Ou seja, se um valor é aleatoriamente escolhido, é imediatamente retirado do vector V e o valor i passa de $i = 1, 2, \dots, n$ para $i = 1, 2, \dots, n - 1$. Este processo é repetido até que não haja valores armazenados nos vectores. Esta metodologia introduz outra grande vantagem, não é necessário proceder à verificação dos testes de hipótese. Como todas as variáveis são provenientes da mesma função de distribuição e não podem ser repetidas, não há necessidade de aplicar os testes de hipótese. Ou seja se for retirado um valor correspondente a cada centro de massa de todas as fatias de uma função densidade de probabilidade, a amostra resultante terá exactamente as mesmas características que a função teórica que a define, ao contrario do método de Monte Carlo.

O método do Hipercubo Latino é geralmente usado em sistemas dependentes de múltiplas variáveis aleatórias. Na figura 2.12 representa-se um modelo simplificado de um sistema com duas variáveis aleatórias, que descreve o método do Hipercubo Latino. Como é possível observar, temos 6 áreas para cada variável, o que corresponde a seis valores por variável, X e Y , sendo cada valor utilizado uma única vez. Os pontos pretos representam o número de simulações e o centro de gravidade combinado de duas fatias. Na tabela 2.4 exemplifica-se a geração de valores aleatórios para o caso apresentado na figura 2.12.

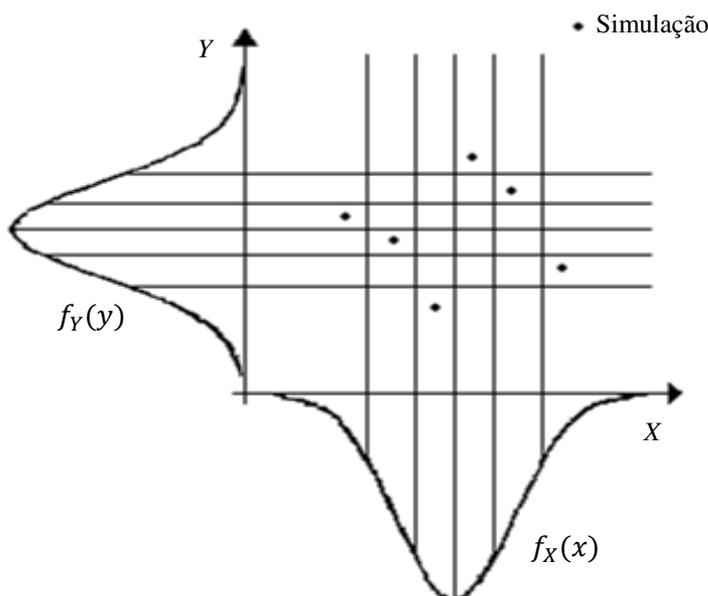


Fig. 2.12 - Representação gráfica da segunda fase do método do Hipercubo Latino

Na tabela 2.4 pode-se ver seis colunas distintas: na primeira demonstra-se o número de simulações efectuadas; a segunda contém uma variável, à qual lhe é somada uma unidade no fim de cada simulação. Esta variável é subtraída a n , número total de simulações, para representar o decréscimo de variáveis nos vectores; a terceira e quarta coluna representam a aplicação do algoritmo gerador de valores aleatórios às variáveis aleatórias X e Y , respectivamente; nas últimas duas colunas estão dispostos os valores das variáveis, X e Y , que vão sendo aleatoriamente escolhidos. Analisando a terceira e quarta coluna pode-se observar que existem números que são repetidos. Por exemplo na primeira simulação o algoritmo aleatório gera os valores 3 e 1, sendo escolhidos e retirados do vector, os valores X_3 e Y_1 . Na simulação seguinte o algoritmo gera 4 e 4, sendo escolhidos, X_5 e Y_5 . Como os X_3 e Y_1 já foram retirados do vector, os valores X_4 e Y_2 tomam os seus lugares, sendo uma reacção em cadeia até aos valores X_6 e Y_6 que tomam a posição 5 e 5 nos respectivos vectores.

Tabela 2.4 – Resultado da simulação efectuada na figura 2.12

Nº de Simulações	j	$i_x = \text{int} [\text{rand}(n - j)] + 1$	$i_y = \text{int} [\text{rand}(n - j)] + 1$	X	Y
1	0	3	1	X_3	Y_1
2	1	4	4	X_5	Y_5
3	2	1	3	X_1	Y_4
4	3	1	2	X_2	Y_3
5	4	2	1	X_6	Y_2
6	5	1	1	X_4	Y_6
-	6	0	0	-	-

3

PROGRAMAÇÃO

3.1. INTRODUÇÃO

Com os conhecimentos de estatística, avaliação de fiabilidade estrutural e métodos de simulação adquiridos procede-se à elaboração do programa de cálculo automático de apoio à avaliação de segurança de elementos estruturais. Este programa está dividido em dois módulos: No primeiro é efectuada a simulação de amostras aleatórias pelo método de Monte Carlo ou pelo método do Hipercubo latino; No segundo é efectuada a avaliação de fiabilidade estrutural por cálculo indirecto da probabilidade de rotura. Entre todos os “softwares” de programação existentes no mercado, o mais indicado para esta dissertação é o “Matlab”. O mesmo, é usado frequentemente para aplicações de estatística, tendo por isso uma vasta biblioteca de fórmulas que facilitam a operação de cálculos estatísticos (como por exemplo as funções de distribuição estatística).

No presente capítulo será exposto o funcionamento dos dois módulos do programa de cálculo de apoio à avaliação estrutural, o funcionamento interno dos algoritmos desenvolvidos na dissertação e ainda será efectuado um exemplo de aplicação num problema simples de engenharia civil.

O programa foi desenvolvido através da plataforma “Matlab” utilizando uma interface gráfica interactiva. Desta forma desenvolveu-se uma ferramenta muito intuitiva e de fácil utilização. Os dois módulos foram elaborados de forma a serem independentes, ou seja o utilizador pode aceder às funções do segundo módulo sem preencher os espaços no primeiro módulo. Assim tem-se uma ferramenta flexível e útil, podendo ser utilizada em problemas que não estejam no âmbito da avaliação de fiabilidade estrutural (principalmente o primeiro módulo).

O utilizador ao abrir o programa abre uma janela onde pode escolher opções e introduzir dados numéricos e alfabéticos. Adicionados os dados pretendidos é possível exportar as simulações efectuadas para o programa “Excel”, onde podem ser tratadas. Após tratamento, com o vector “resistente”, R, e o vector “acções”, S, calculados, é possível importar dados e calcular indirectamente (índice de fiabilidade) a respectiva probabilidade de rotura. Para uma melhor análise de resultados, são disponibilizados histogramas dos três sistemas, resistente, acção e de resposta.

Na figura 3.1 está representado o aspecto inicial da interface gráfica do programa de apoio. Como é possível observar a janela está dividida em dois, sendo a parte superior destinada à simulação de amostras e a inferior destinada ao cálculo de probabilidades de rotura.

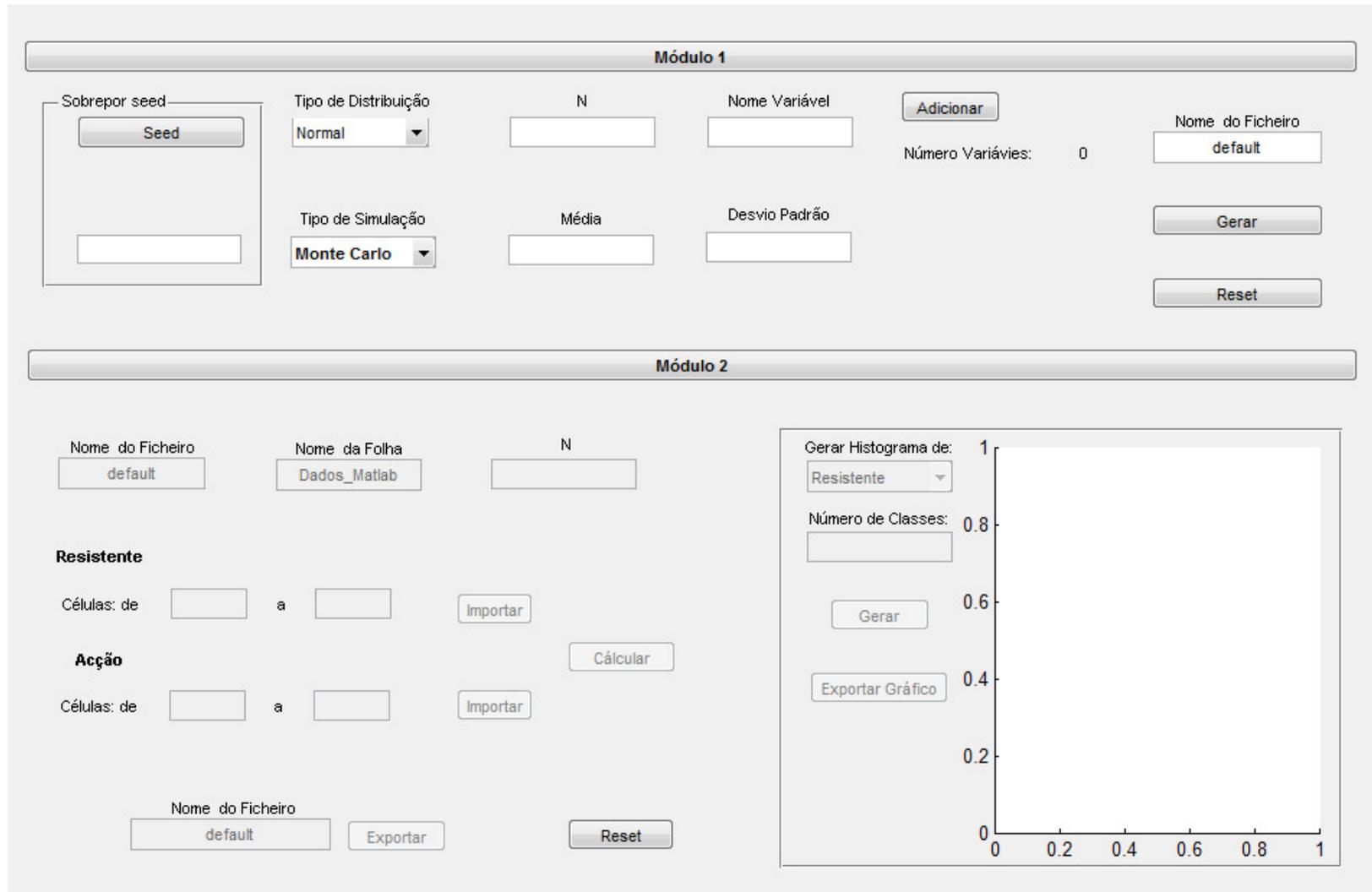


Fig. 3.1 - Interface gráfica do programa de apoio à avaliação de fiabilidade estrutural

3.2. PRIMEIRO MÓDULO – SIMULAÇÃO DE AMOSTRAS

Como já foi referido atrás, neste módulo vão ser efectuadas as simulações de acordo com o método de Monte Carlo e Hipercubo Latino. Neste ponto serão expostos os passos para obter as amostras e explicações dos algoritmos envolvidos.

Antes de começar a introduzir dados no programa é necessário decidir o modo de geração da seed. Tem-se duas opções, deixar o programa retirar uma seed que depende de um parâmetro temporal, ou seja pseudo-aleatório, ou então estabelece-se uma seed inicial. Para tal basta introduzir no espaço editável a seed desejada e carregar no botão “seed”. Deste modo é possível reproduzir o mesmo conjunto de amostras com a mesma ordem em simulações diferentes.

Começa-se por tomar duas decisões, quantas amostras são pretendidas e qual vai ser o método de simulação utilizado. Deve-se proceder a esta escolha porque após a adição da primeira variável aleatória estes dois parâmetros ficam bloqueados. Não faz sentido fazer amostragens de múltiplas variáveis aleatórias com diferentes tamanhos e também não faz sentido misturar os dois métodos de simulação. Assim, evita-se erros que o utilizador possa introduzir involuntariamente.

De seguida procede-se à caracterização da primeira variável, definindo o nome da variável e os seus parâmetros de distribuição. Os parâmetros variam de distribuição para distribuição, mas como mostra a figura 3.2, existem caixas de texto para ajudar a compreender, que mudam de acordo com a distribuição que se escolhe na lista “tipo de distribuição”. É necessário ter especial cuidado, em compreender a lógica de cada parâmetro, pois são estes que melhor caracterizam uma função de distribuição.

The figure displays two screenshots of a software interface for simulation setup, labeled 'Módulo 1' and 'Módulo 2'.

Módulo 1 (Top Screenshot):

- Tipo de Distribuição:** Normal (selected in dropdown menu)
- N:** 10000
- Nome Variável:** A
- Adicionar:** Button
- Número Variáveis:** 0
- Média:** 896
- Desvio Padrão:** 78.45

Módulo 1 (Bottom Screenshot):

- Tipo de Distribuição:** LogNormal (selected in dropdown menu)
- N:** 10000
- Nome Variável:** B
- Adicionar:** Button
- Número Variáveis:** 1
- Mu:** 1
- Sigma:** 0.5
- Média da distribuição normal associada:** 1
- Desvio padrão da distribuição normal associada:** 0.5

Fig. 3.2 - Imagem retirada do programa

Existem duas funções de distribuição que merecem especial atenção, extremos tipo II e extremos tipo III. Ao seleccionar uma destas duas distribuições surge uma nova caixa editável que solicita um valor que caracterize a forma da distribuição, “ K ”. O utilizador deve preencher este espaço com valores superiores a zero se pretender uma distribuição do tipo II e deve preencher com valores inferior a zero se pretender uma distribuição do tipo III. Na figura 3.3 estão representados os dois tipos de distribuições de extremos com três parâmetros “ K ”. As seis curvas têm parâmetros de localização e escala iguais.

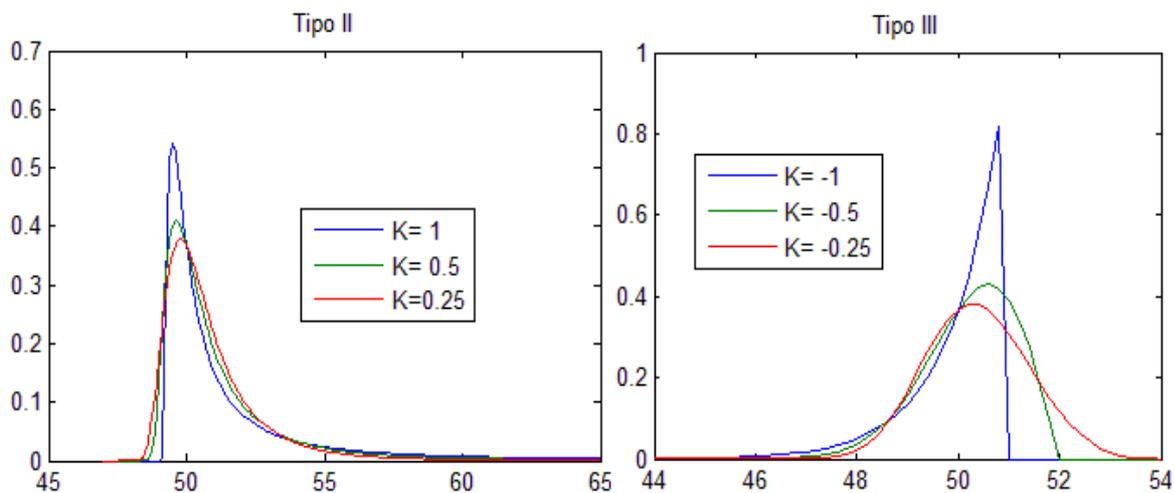


Fig. 3.3 - Caracterização do parâmetro K das leis do tipo II (à esquerda) e III (à direita)

Ao carregar no botão “Adicionar” (à direita) o programa vai correr uma fórmula da função de distribuição acumulada inversa da distribuição escolhida, com os parâmetros que estão nos espaços editáveis. A simulação será feita de acordo com as secções 2.3.2.1 e com 2.3.3.1. Os valores gerados das variáveis são armazenados numa matriz, onde cada coluna corresponde a simulação de uma variável. Na figura 3.2 está representado o preenchimento dos espaços editáveis para a primeira e segunda variável, é de notar que o espaço onde se insere o número de simulações está bloqueado (como foi referido anteriormente) e que existe um contador de variáveis.

Após gerar as amostras aleatórias o programa testa-as, efectuando o teste de hipótese. Estes testes só são realizados para as distribuições de variáveis contínuas e se o método de Monte Carlo estiver seleccionado, pois não faz sentido realizá-los para o método do Hipercubo Latino. Se o “tipo de distribuição” for “Normal”, realiza-se primeiro o teste de normalidade e só se este não verificar ou ficar em dúvida, é que se verifica o teste de Kolmogorov-Smirnov. Para os restantes “tipos de distribuição”, procede-se directamente ao segundo teste. Se uma simulação não passar nos testes de hipóteses é repetida até passar. Dentro dos resultados possíveis tem-se: passar no teste de normalidade sem ser submetido ao teste de Kolmogorov-Smirnov; ficar em dúvida no teste de normalidade e passar no teste de Kolmogorov-Smirnov. Nas figuras 3.4 e 3.5 estão representadas duas caixas de mensagens que informam sobre o resultado dos testes de hipóteses.

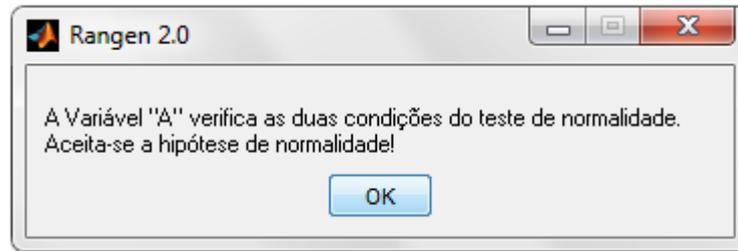


Fig. 3.4 - Exemplo de caixa de mensagem com informação sobre os resultados dos testes de hipótese

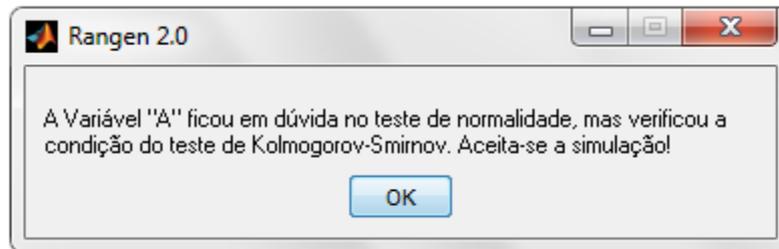


Fig. 3.5 - Exemplo de caixa de mensagem com informação sobre os resultados dos testes de hipótese

Com todas as variáveis aleatórias adicionadas ao programa, gera-se um ficheiro em Excel carregando no botão “Gerar”. O nome predefinido deste ficheiro é “default”, mas pode ser mudado preenchendo o espaço editável que está imediatamente à esquerda do mesmo botão. Ao gerar o ficheiro, o programa cria automaticamente duas folhas: “Dados_Matlab”, onde está guardada a matriz com os valores aleatórios simulados e respectivos nomes; “Teste_de_Hipóteses”, onde estão guardados os resultados dos testes de hipótese, por variável. Nas tabelas 3.1 e 3.2 estão representadas as duas folhas para um caso simples com 5 simulações e 5 variáveis. A tabela 3.1 representa o conteúdo da folha “Dados_Matlab” e tem a primeira coluna com o número identificador da simulação e as restantes com as simulações e os respectivos nomes.

Tabela 3.1 - Exemplo de tabela exportada para a folha "Dados_Matlab"

N	A	B	C	D	E
1	77,94349	24,77599	33,73976	70,25979	99,10922
2	116,5932	93,04316	15,61568	66,4351	116,3467
3	95,8441	74,0908	43,37862	54,11791	118,856
4	94,27807	46,023	37,5914	82,03443	80,29598
5	79,27067	69,67139	34,65677	69,4518	109,0506

A tabela 3.2 representa o conteúdo da folha “Teste_de_Hipóteses”. As duas primeiras linhas estão preenchidas pela legendagem dos dois testes de hipóteses. Nas linhas seguintes apresentam-se os resultados dos testes de hipótese para as cinco variáveis. É de notar que os valores exportados para a linha do teste de Kolmogorov-Smirnov, quando aplicável, são os coeficientes de distribuição, \tilde{D}_n , da equação (2.97). Na última linha apresenta-se o valor da seed inicial usada para a geração de valores aleatórios. Como já foi referido, com este valor é possível reproduzir valores iguais aos obtidos nesta simulação.

Tabela 3.2 - Exemplo de uma tabela exportada para a folha "Testes_de_Hipóteses"

	Teste de Normalidade	a - aceite	d - dúvida	n - não aplicável	r - rejeitado
	Teste de Kolmogorov-Smirnov	valor positivo [0;1,364]-verifica	-2 - Não verifica	-1 - Não aplicável	
Variáveis	A	B	C	D	E
Teste Normalidade	a	d	a	a	a
K-S	-1	0,683856155	-1	-1	-1
seed	10000				

Com os dados exportados para o Excel o utilizador procede ao tratamento de dados utilizando o programa que quiser. Quando esse tratamento estiver pronto o utilizador, se assim pretender, pode utilizar o segundo módulo para efectuar a análise de fiabilidade estrutural.

3.3. SEGUNDO MÓDULO – AVALIAÇÃO DA FIABILIDADE ESTRUTURAL

O segundo módulo não tem uma aplicação tão ampla. Inicialmente, o programa necessita da introdução de 2 conjuntos de dados. O utilizador começa por colocar os dados que pretende avaliar numa folha do tipo Excel. De seguida, deve começar por introduzir nas áreas editáveis, o nome do ficheiro e da folha em que estão armazenados e ainda o número de elementos que cada coluna irá ter. O segundo conjunto de dados a introduzir refere-se à posição inicial e final de cada coluna que contém dados (células de início e de fim).

Com as células definidas procede-se à importação de dados carregando nos dois botões “Importar”. Estes botões foram concebidos independentemente, pois os dados do sistema resistente e das acções solicitadas podem estar em folhas ou ficheiros diferentes. Ao importar estes dados o programa vai verificar a normalidade dos dois campos através dos testes de hipótese. Neste módulo efectua-se o teste de normalidade e o teste de Kolmogorov-Smirnov modificado por Lilliefors, pois os parâmetros da lei probabilística a testar são estimados a partir da amostra. Ao contrário do módulo anterior, neste aceitam-se sistemas resistentes e acções que não verifiquem os testes de hipótese. Na engenharia civil é usual encontrar sistemas que não se comportam segundo uma distribuição normal. Sendo assim podem ser dispostas quatro mensagens diferentes: verifica o teste de normalidade (exemplo figura 3.6); não verifica a segunda condição do teste de normalidade, mas verifica o teste de Kolmogorov-Smirnov; não verifica o teste de normalidade, mas verifica o teste de Kolmogorov-Smirnov; não verifica nenhum dos dois testes.

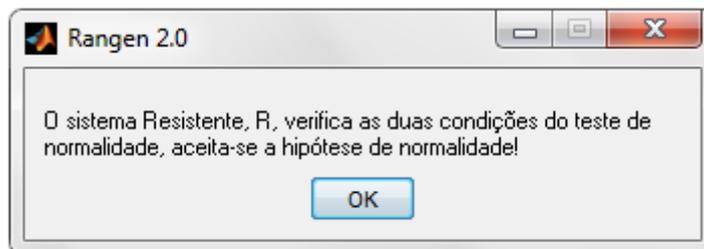


Fig. 3.6 - Exemplo de mensagem de verificação dos testes de hipótese do segundo módulo

Segue-se o cálculo da probabilidade de rotura através do cálculo do índice de fiabilidade. Ao carregar no botão “Calcular”, o programa calcula as médias e desvios padrão do sistema resistente e das acções. De seguida calcula o índice de fiabilidade e a respectiva probabilidade de rotura pelo procedimento proposto na secção 2.2.5.2 desta dissertação. Em conjunto com este procedimento o programa também determina o erro associado e a COV segundo a secção 2.3.2.3 desta dissertação. Depois de calculados, pode-se exportar os dados obtidos da mesma forma como se fez no primeiro módulo. A tabela 3.3 é um exemplo dos dados exportados. Nas duas primeiras linhas, da folha “Resultados_Matlab”, temos o resultado da probabilidade de rotura, índice de fiabilidade, COV e do erro (mesma ordem da tabela). Na quarta e quinta linha tem-se os resultados dos dois testes de hipóteses. A legenda é igual à do primeiro módulo. Na primeira coluna, a partir da sétima linha, tem-se os valores da resposta do sistema estrutural.

Tabela 3.3 - Exemplo de resultados exportados para a folha "Resultados_Matlab", no segundo módulo

Pf	Beta	COV	Erro(%)
0,030500929	1,873481991	0,182841468	35,65719393
Teste de norm	a	A	
K-S	-1	-1	
Z			
24,45146632			
18,78271173			
...			
Zn			

Com a intenção de melhorar e ajudar na análise do comportamento de um sistema estrutural, o programa tem a capacidade traçar histogramas com uma curva teórica ajustada do sistema resistente, da acção solicitada e ainda da resposta do sistema estrutural. Podem ser escolhidas o número de classes que se pretende, introduzindo no espaço editável, como demonstra a figura 3.7. Se for necessário, é possível exportar o gráfico da figura 3.7. Ao carregar no botão “Exportar Gráfico”, abre uma janela com o mesmo gráfico, na qual é possível editar a área do gráfico (como por exemplo adicionar pequenas notas).

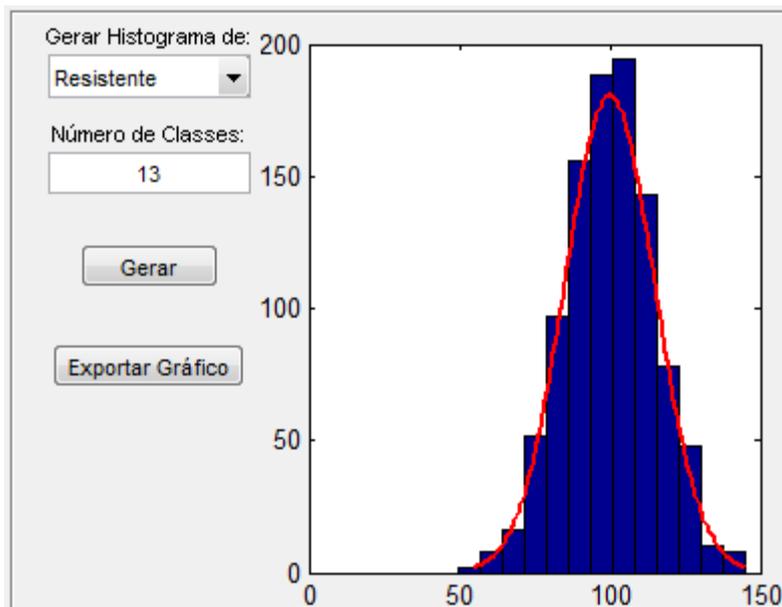


Fig. 3.7 - Interface gráfica com representação do histograma

3.4. EXEMPLO DE APLICAÇÃO

Para melhor compreender as várias funcionalidades do programa de apoio à avaliação da fiabilidade estrutural, apresenta-se um exemplo de aplicação simples. Este exemplo consiste no cálculo do momento resistente de uma secção de betão armado, à tracção, e na verificação da probabilidade de rotura quando este é solicitado por uma acção. Tudo isto será efectuado por fases:

- começando pela caracterização estatística das variáveis aleatórias;
- geração de amostras aleatórias recorrendo ao programa de apoio (módulo 1);
- cálculo do momento resistente e do momento actuante, numa folha de Excel;
- acabando com a avaliação da fiabilidade estrutural com ajuda do programa (módulo2).

Este procedimento será repetido para os dois métodos de simulação estudados nesta dissertação e ainda para simulações de 100, 1000, 5000 e 10000 amostras. De modo a não se tornar demasiado exaustivo, vai-se exemplificar o procedimento para 100 simulações através do método de Monte Carlo e no fim apresenta-se os restantes resultados.

É importante referir que para o método de Monte Carlo, não se deve utilizar 100 simulações para fins de avaliação de fiabilidade estrutural. Para tamanho de amostras pequenas este método é muito impreciso, como se vai demonstrar mais a frente. Neste exemplo será efectuado uma simulação de 100 amostras, somente para comparar com os resultados do método do Hipercubo Latino.

3.4.1. GEOMETRIA E CARACTERIZAÇÃO DAS VARIÁVEIS ALEATÓRIAS

Considere uma viga com um vão de cinco metros, apoiada nas duas extremidades, com secção de betão C25/30 de 30cm por 60cm, reforçado com aço S500. A armadura inferior pelo menos com 1% da área da secção e a armadura superior com 20% da armadura inferior. Na figura 3.8 estão representadas as características da viga e a carga, q , que será 100kN/m inicialmente e 130kN/m numa segunda fase.

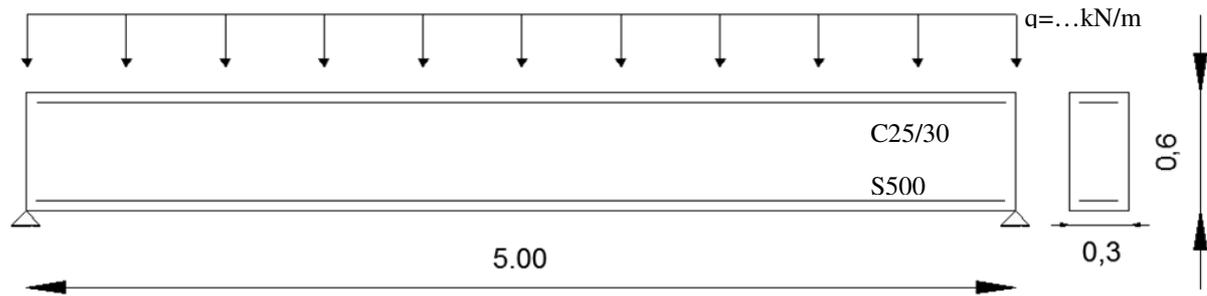


Fig. 3.8 - Geometria da viga

Usando valores determinísticos e sabendo que a viga tem uma secção de $0.30 \times 0.60 \text{ m}^2$, chega-se a conclusão que é necessário colocar 18 cm^2 de armadura inferior e 3.6 cm^2 de armadura superior. Então para a armadura inferior aplicam-se 6 varões de 20mm ($6\phi 20$) e para a superior 2 varões de 16mm ($2\phi 16$).

Como já foi referido na secção 2.3.2, após gerar amostras aleatórias, calcula-se, deterministicamente, o momento resistente para cada uma das amostras geradas. Este cálculo é efectuado segundo o EC-0 (2002), parte 1-1, na secção 3.1.7 (3). Começa-se por calcular a posição do eixo neutro, x , pelo equilíbrio de forças representado na figura 3.9 e na expressão 3.1.

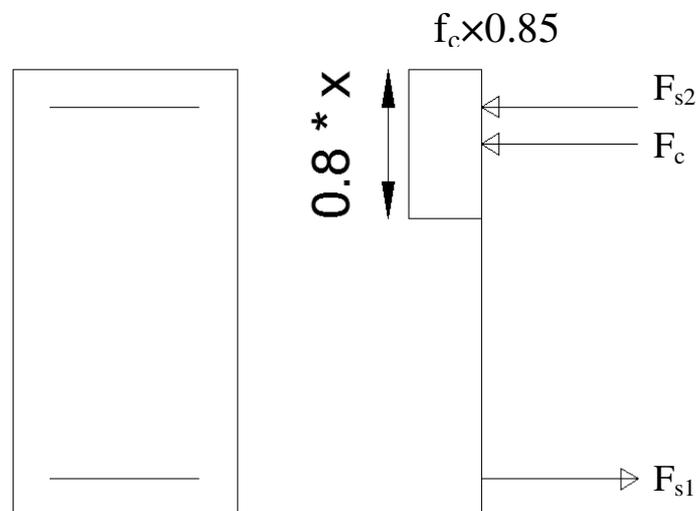


Fig. 3.9 - Representação gráfica do equilíbrio de forças

$$F_{s1} = F_c + F_{s2} \quad , \quad (3.1)$$

onde F_{s1} é a força resultante da armadura inferior, F_c é a força resultante do betão e F_{s2} é a força resultante do armadura superior.

Antes de começar com os cálculos efectivos, convém conhecer quais são as variáveis aleatórias necessárias para calcular o momento resistente e conhecer os respectivos parâmetros estatísticos. As

variáveis e os seus parâmetros de distribuição estatística encontram-se na tabela 3.4 (adaptado de Wiśniewski, 2007).

Tabela 3.4 - Parâmetros de distribuição das variáveis envolvidas

Variável Aleatória	Descrição	Tipo de distribuição	Média	Desvio padrão
Betão				
f_c	- Resistência do betão à compressão C25/30	Normal	37.8 MPa	2.9106 MPa
d	- Altura útil	Normal	55 cm	1.1 cm
b	- Largura da viga	Normal	30 cm	0.6 cm
c	- Recobrimento	Normal	5 cm	0.1 cm
L	- Comprimento da viga	Normal	500 cm	5 cm
Aço				
f_{syi}	- Tensão de cedência da armadura inferior (S500 Ø20)	Normal	594 MPa	31 MPa
A_{si}	- Área da armadura inferior (Ø20)	Normal	2.96 cm ²	0.09 cm ²
f_{sys}	- Tensão de cedência da armadura inferior (S500 Ø16)	Normal	601 MPa	35 MPa
A_{ss}	- Área da armadura superior (Ø16)	Normal	1.83 cm ²	0.08 cm ²

Antes de começar com a simulação de amostras é necessário definir a fórmulas que serão aplicadas para calcular a posição do eixo neutro e o valor do momento resistente:

- Expressão de equilíbrio,

$$f_{syi} \times 6 \times A_{si} = f_{sys} \times 2 \times A_{ss} + f_c \times 0.8 \times b \times x . \quad (3.2)$$

- Então tem-se o x:

$$x = \frac{f_{syi} \times 6 \times A_{si} - f_{sys} \times 2 \times A_{ss}}{f_c \times 0.8 \times b} . \quad (3.3)$$

- Equação do momento resistente em relação à face superior (figura 3.10):

$$M_R = F_{s1} - F_c - F_{s2} , \quad (3.4)$$

- decompondo as forças:

$$M_R = d \times f_{syi} \times 6 \times A_{si} - c \times f_{sys} \times 2 \times A_{ss} - \frac{0.8 \times x}{2} \times f_c \times b \times 0.8 \times x . \quad (3.5)$$

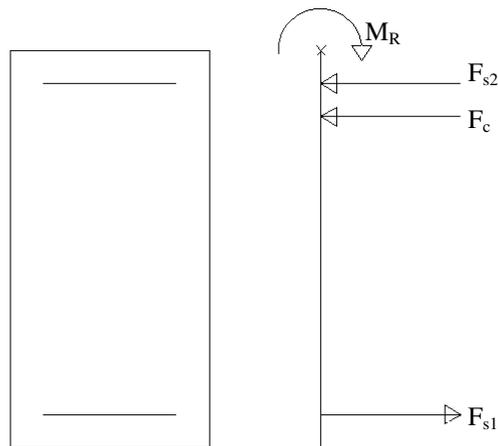


Fig. 3.10 - Representação gráfica da equação do momento resistente em relação à face superior

3.4.2. SIMULAÇÃO DE AMOSTRAS

Com as variáveis aleatórias estatisticamente caracterizadas e fórmulas estabelecidas, procede-se à simulação de amostras. Começa-se por simular 100 amostras. Se o erro for grande demais, repete-se com um número superior até que se encontre um erro razoável.

Após gerar as amostras, o utilizador pode escolher a forma como quer efectuar os cálculos. Neste caso os cálculos foram efectuados na mesma folha para a qual foram exportados. Basta adicionar duas colunas com fórmulas e obtém-se os valores do eixo neutro e do momento resistente.

Se o utilizador tiver as fórmulas preparadas no ficheiro Excel, evita-se copiar valores para outros programas, evitando erros por negligência. A tabela 3.5 contém a média e desvio padrão da posição do eixo neutro, x , e do momento resistente, M_R . A mesma tabela contém ainda os resultados dos testes de hipótese para todas as variáveis simuladas.

Tabela 3.5 - Resultados da simulação e dos testes de hipótese, para 100 amostras

	M_R					x			
Média	538.134					0.0922			
Desvio Padrão	29.0115					0.0091			
Teste:	f_c	d	b	c	L	f_{syi}	A_{si}	f_{sys}	A_{ss}
Normalidade	a	a	a	a	a	a	a	a	a
K-S	-1	-1	-1	-1	-1	-1	-1	-1	-1

Analisando as duas últimas linhas da tabela 3.5, conclui-se que todas as simulações verificaram o teste de normalidade, a , e não foram sujeitas ao teste de Kolmogorov-Smirnov, -1 .

Falta agora calcular a acção. Como foi dito antes, a estrutura vai ser solicitada por duas cargas diferentes. Considera-se uma carga de $100kN/m$ e outra de $130kN/m$, com peso próprio da viga incluído. Para simplificar os cálculos, considera-se que a relação entre a média da acção solicitada e o seu desvio padrão, é igual à mesma relação do momento resistente. Então:

$$\frac{\sigma_R}{\mu_R} = \frac{29.01149}{538.1335} = 0.05391 \quad , \quad (3.6)$$

$$0.05391 \times \mu_S = \sigma_S = 5.391kN/m \quad , \quad (3.7)$$

sendo $\mu_S = 100kN/m$ e :

$$0.054 \times \mu_S = \sigma_S = 7.008kN/m \quad , \quad (3.8)$$

sendo $\mu_S = 130kN/m$.

De seguida simula-se as variáveis aleatórias que representam as acções solicitadas e inicia-se o segundo módulo. O momento actuante calcula-se pela fórmula do momento máximo num elemento simplesmente apoiado, $p \times l^2/8$. Para $q = 100 kN/m$ tem-se um momento actuante com média de $310.603kN/m$ e desvio padrão igual a $17.976kN/m$. Para $q = 130 kN/m$ tem-se momento actuante com média de $406.64kN/m$ e desvio padrão igual a $21.29kN/m$ (Tabela 3.6).

3.4.3. AVALIAÇÃO DE FIABILIDADE ESTRUTURAL

Introduzindo os valores do momento resistente e das duas cargas solicitadas, o programa calcula o índice de fiabilidade e a probabilidade de rotura. Os cálculos são efectuados de acordo com a secção 2.2.5 desta dissertação. Os resultados para 100 simulações pelo método de Monte Carlo e para as duas cargas solicitadas encontram-se nas tabelas 3.6 e 3.7.

Tabela 3.6 - Momentos e resposta estrutural para 100 amostras

Método Monte Carlo N=100									
MR		MS – 100kN/m		MS – 130kN/m		Z – 100kN/m		Z – 130kN/m	
μ_R	σ_R	μ_S	σ_S	μ_S	σ_S	μ_Z	σ_Z	μ_Z	σ_Z
538.13	29.01	301.60	17.98	406.64	21.29	227.53	34.92	131.50	36.73
Teste de normalidade - Dúvida		Teste de normalidade - Verifica		Teste de normalidade - Verifica					
K-S = 0.6325		K-S Não necessário		K-S Não necessário					

Tabela 3.7 - Resultados da amostragem de 100 simulações

Método Monte Carlo N=100							
Para q=100kN/m				Para q=130kN/m			
β	Pf	COV	$\epsilon(\%)$	β	Pf	COV	$\epsilon(\%)$
6.516	6.61E-11	-	3328539	3.580	1.72E-4	-	1524

Os dados introduzidos no segundo módulo são sempre sujeitos a um teste de normalidade e se for necessário, ao teste de Kolmogorov-Smirnov. Segundo os dados obtidos, expostos na tabela 3.6, o momento resistente só verificou a segunda condição do teste de normalidade, mas verificou o teste de Kolmogorov-Smirnov. Os momentos actuantes também são sujeitos a estes testes, mas como

verificam o teste de normalidade não necessitam de verificar o segundo teste. O primeiro facto que se faz notar na tabela 3.7, é o valor dos erros. Comparando os erros com o valor aceitável, na ordem dos 10%, pode concluir-se que é necessário aumentar, largamente, o número de simulações efectuadas. Outro resultado que pode parecer estranho, é a ausência de valores na coluna das *COV*. Isto deve-se ao facto de o cálculo deste coeficiente ter por base a contagem do número de roturas no campo amostral. Assim pode-se concluir que em 100 amostras não houve uma única rotura.

Tendo em conta a tabela 2.2 com os valores recomendados para índices de fiabilidade verifica-se que para a carga inicial, a viga encontra-se em segurança para todas as classes de fiabilidade e para os dois períodos de referência. Para a segunda carga aplicada, verifica-se que só se respeita a classe de fiabilidade RC-1 para um período de referência de 50 anos (limite $\beta=3.3$).

De seguida apresentam-se os histogramas com curva teórica ajustada, exportados a partir do programa.

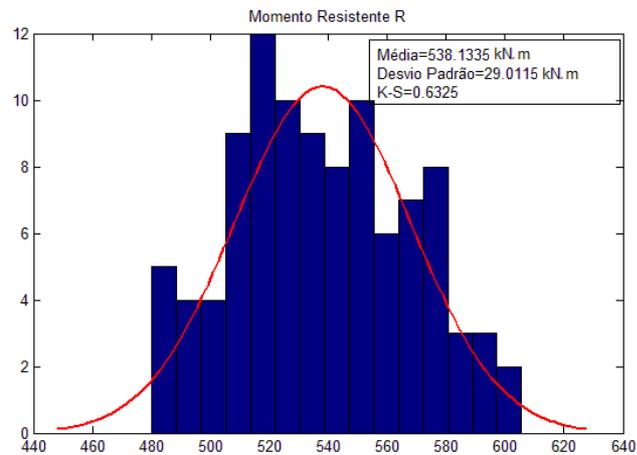


Fig. 3.11 - Histograma com curva teórica ajustada do momento resistente para 100 amostras

Na figura 3.11 está representado o histograma do momento resistente para a simulação de 100 amostras. É de notar que a distribuição afasta-se ligeiramente da curva teórica. Tal distribuição deve-se ao facto de estes valores terem sido sujeitos a cálculos e ao baixo numero de simulações. Com um número maior de simulações, a distribuição do momento resistente deverá aproximar-se da curva teórica. Analisando as restantes figuras, constata-se o mesmo facto, distribuição ligeiramente afastada da curva teórica. Procede-se então aos cálculos para um número superior de simulações e para o outro método de simulação estudado nesta dissertação, método do Hipercubo Latino.

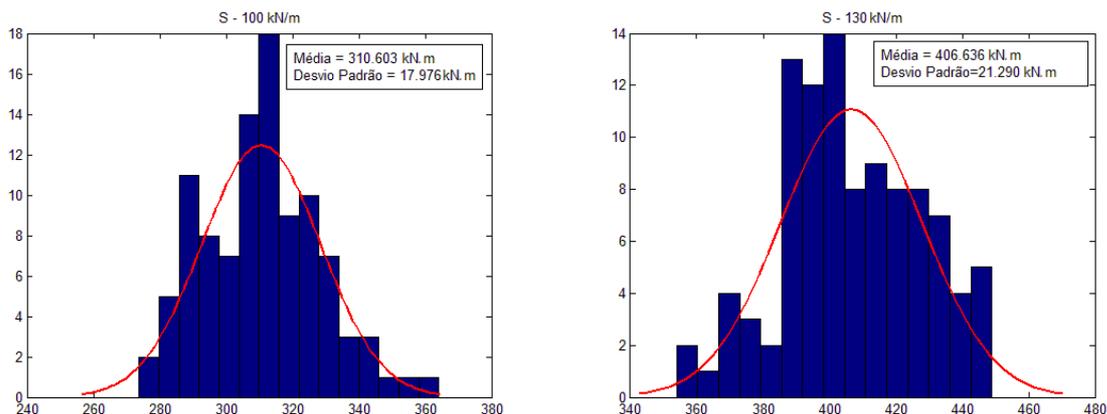


Fig. 3.12 - Histogramas com curva teórica ajustada dos momentos actuantes para 100 amostras, quando $q=100\text{kN/m}$ (à esquerda) e $q=130\text{kN/m}$ (à direita)

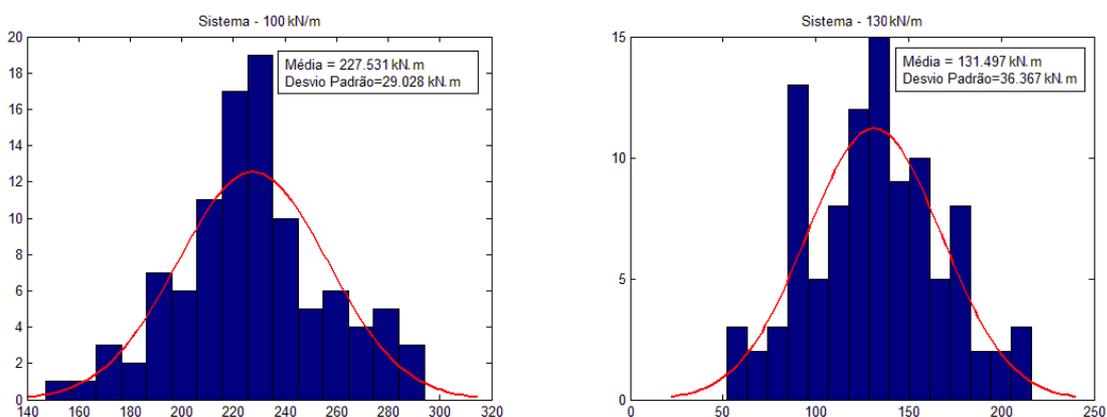


Fig. 3.13 - Histogramas com curva teórica ajustada das respostas estruturais para 100 amostras, quando $q=100\text{kN/m}$ (à esquerda) e $q=130\text{kN/m}$ (à direita)

3.4.4. SIMULAÇÃO E AVALIAÇÃO DE FIABILIDADE ESTRUTURAL – MÉTODO DE MONTE CARLO

De seguida serão expostos os resultados obtidos na avaliação de fiabilidade estrutural pelo método de Monte Carlo, para 1000, 5000 e 10000 simulações. Começa-se por simular as amostras para todas as “N” simulações.

Tabela 3.8 - Resultados dos testes de Hipóteses para 1000 amostras

Monte Carlo N=1000									
Teste:	f_c	d	b	c	L	f_{syi}	A_{Si}	f_{sys}	A_{SS}
Normalidade	a	a	a	d	a	a	a	a	d
K-S	-1	-1	-1	1,104632088	-1	-1	-1	-1	0,537461

Tabela 3.9 - Resultados dos testes de Hipóteses para 5000 amostras

Monte Carlo N=5000									
Teste:	f_c	d	b	c	L	f_{syi}	A_{si}	f_{sys}	A_{ss}
Normalidade	a	d	a	d	a	a	a	d	a
K-S	-1	0,699313	-1	0,828673	-1	-1	-1	0,796416	-1

Tabela 3.10 - Resultados dos testes de Hipóteses para 10000 amostras

Monte Carlo N=10000									
Teste:	f_c	d	b	c	L	f_{syi}	A_{si}	f_{sys}	A_{ss}
Normalidade	a	a	d	a	a	d	a	a	d
K-S	-1	-1	0,932952	-1	-1	0,643043	-1	-1	0,7592

Analisando as tabelas Tabela 3.8, Tabela 3.9 e Tabela 3.10, constata-se que existem em todas as “N” simulações, algumas variáveis aleatórias que não verificam as primeira condição do teste de normalidade, mas verificam o teste de Kolmogorov-Smirnov. Procede-se agora ao cálculo dos momentos solicitadores e ao cálculo da resposta do sistema estrutural para todas as “N” simulações.

Tabela 3.11 – Momento resistente e momentos actuantes para 100, 1000, 5000 e 10000 simulações

Método Monte Carlo						
N	$M_R(\text{kN.m})$		$M_S(\text{kN.m}) - 100\text{kN/m}$		$M_S(\text{kN.m}) - 130\text{kN/m}$	
	μ_R	σ_R	μ_S	σ_S	μ_S	σ_S
100	538.134	29.011	310.603	17.976	406.636	21.290
1000	538.131	33.201	313.038	19.628	406.319	27.2959
5000	537.657	32.877	312.437	19.924	406.172	26.157
10000	538.098	32.184	312.481	19.726	405.987	25.702

Numa primeira análise da tabela 3.11, pode-se afirmar que, em relação a 100 simulações, as médias não variaram muito, por outro lado, os desvios padrão tiveram uma variação considerável, sendo para $q=130\text{kN/m}$ superior a 20%. Introduzindo os dados obtidos, no programa resultam os valores da resposta estrutural, que se encontram nas tabelas 3.12 e 3.13. Para os testes de hipóteses, realizados ao introduzir os momentos no programa obtêm-se valores diferentes em todas as “N” simulações, obtendo também uma não verificação do teste de normalidade, para 10000 simulações. Contudo para o mesmo número de simulações verifica-se o teste de Kolmogorov-Smirnov.

Tabela 3.12 - Resultados dos testes de hipótese no segundo módulo, para 1000, 5000 e 10000 amostras

Método Monte Carlo						
	1000		5000		10000	
Teste de Hipótese	Norm.	K-S	Norm.	K-S	Norm.	K-S
$M_R(\text{kN.m})$	d	0.835	a	-	r	0.645
$M_S(\text{kN.m}) - 100\text{Kn/m}$	a	-	a	-	a	-
$M_S(\text{kN.m}) - 130\text{Kn/m}$	a	-	a	-	a	-

À semelhança dos momentos resistentes e dos actuantes, os momentos resultantes (Tabela 3.13) apresentam uma média semelhante à obtida, a partir das 100 simulações, mas apresentam um desvio padrão com uma variação mais acentuada.

Tabela 3.13 - Resposta estrutural para todas as amostragens

Método Monte Carlo				
	Z – 100kN/m		Z – 130kN/m	
N	μ	σ	μ	σ
100	227.531	34.919	131.497	36.734
1000	225.093	38.569	131.612	42.981
5000	225.223	38.443	131.487	42.013
10000	225.617	37.648	132.111	41.187

Para as 100 simulações, como já foi constatado anteriormente, existe um erro excessivo para as duas cargas. Com o aumento de número de simulações este erro diminuiu consideravelmente. Para $q=130\text{kN/m}$ tem-se um erro final de 77.3%. Ainda não é um erro admissível, mas analisando o decréscimo do erro das 100 simulações para 10000, pode-se concluir que um número maior de simulações obtém-se um erro admissível. Em contrapartida para $q=100\text{kN/m}$, para 10000 simulações o erro está longe de atingir os 10% pretendidos. Tal deve-se ao facto de a probabilidade de rotura ser muito pequena. Recordando a expressão:

$$\varepsilon\% = \sqrt{\frac{(1 - P_f^R)}{N \times P_f^R}} \times 200\% . \tag{2.100}$$

da secção 2.3.2.3, quanto menor é a probabilidade mais simulações são necessárias para obter um erro admissível. Então para atingir um erro desejável seria necessário cerca de 1.8E10 simulações para $q=100\text{kN/m}$ e cerca de 30000 para $q=130\text{kN/m}$. Como é possível observar na tabela 3.14 para a carga inicial não é possível calcular o valor de COV, ou seja de entre todo o espaço amostral não houve um único caso de rotura. Daí a probabilidade de rotura tão baixa. Contudo para a segunda carga e para 5000 simulações, O COV já apresenta valores, o que significa que há alguns casos de rotura.

Tabela 3.14 - Resultados finais para 100, 1000, 5000,10000

Método Monte Carlo								
Para q=100kN/m					Para q=130kN/m			
N	β	Pf	COV	$\epsilon(\%)$	β	Pf	COV	$\epsilon(\%)$
100	6.516	6.61E-11	-	3328539	3.580	1.72E-4	-	1524
1000	5.836	2.67E-09	-	122334.9	3.062	0.001099	-	190.668
5000	5.858	2.33E-09	-	58545.9	3.130	0.000875	0.49978	95.575
10000	5.977	1.14E-09	-	59324.1	3.208	0.000669	0.33322	77.279

Tendo em conta a tabela 2.2 com os valores recomendados para índices de fiabilidade (EC-0, 2002) verifica-se que para a carga inicial a viga encontra-se em segurança para todas as classes de fiabilidade e para os dois períodos de referência. Para a segunda carga aplicada, contrariamente à amostragem de 100, já nem verifica a classe de fiabilidade RC1 para 50 anos de referência. De seguida apresentam-se os histogramas com curvas teóricas ajustadas dos momentos resistentes e das respostas dos sistemas estruturais para 1000, 5000 e 10000 simulações com $q=130kN/m$.

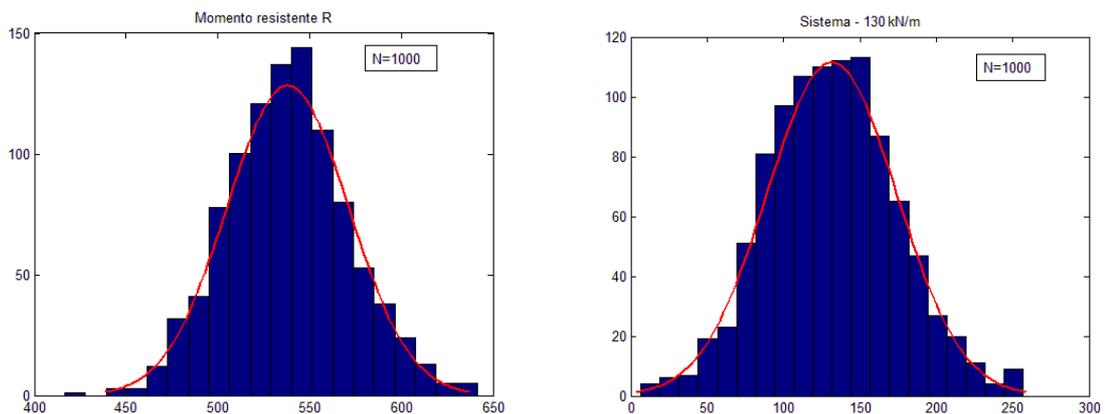


Fig. 3.14 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 1000 amostras

É de notar que para 100 e 1000 simulações os histogramas não apresentam valores negativos. Isto vem confirmar, a inexistência de casos de rotura e os valores obtidos para COV. Em contra partida os histogramas da resposta estrutural para amostragens de 5000 e 10000 já apresentam valores negativos, como é possível confirmar na figura 3.15.

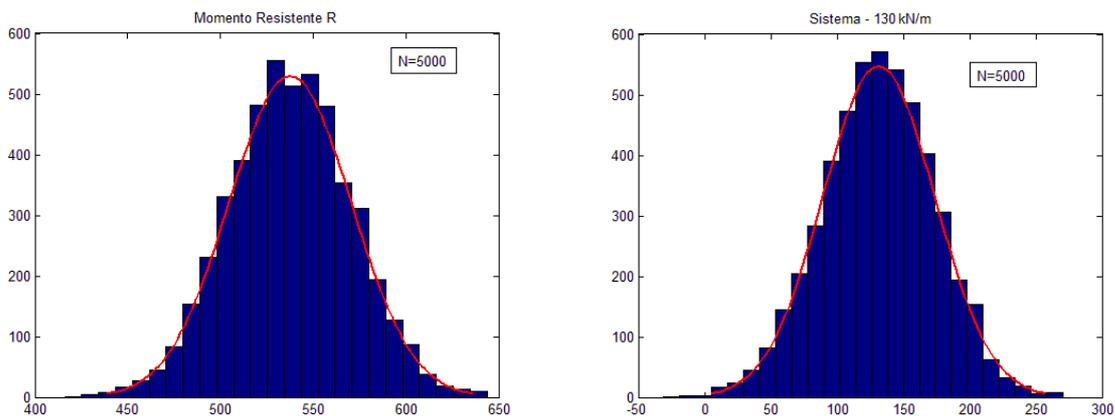


Fig. 3.15 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 5000 amostras

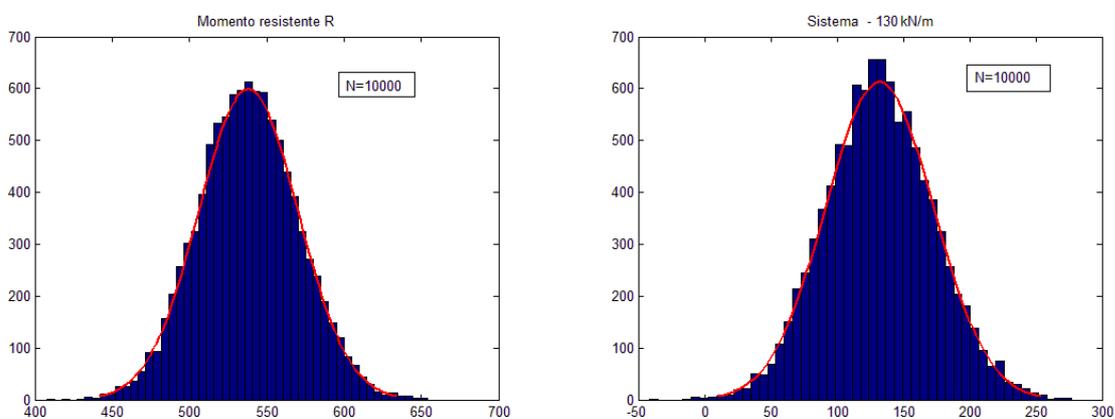


Fig. 3.16 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 10000 amostras

3.4.5. SIMULAÇÃO E AVALIAÇÃO DE FIABILIDADE ESTRUTURAL – MÉTODO DO HIPERCUBO LATINO

Procede-se agora à simulação através do método do Hipercubo Latino e respectivos cálculos. Como já foi referido na secção 2.3.3.1, não é necessário proceder à verificação dos testes de hipótese, quando as amostras são geradas. Então os resultados, relacionados com testes de hipótese, exportados pelo programa no primeiro módulo, serão “não aplicável”. Para este método apresenta-se tabelas finais com valores para todas as “N” simulações. Começa-se por expor os resultados para os momentos resistentes e os momentos actuantes ($q=100kN/m$ e $q=130kN/m$), usando os mesmos parâmetros de distribuição (Tabela 3.15).

Tabela 3.15 - Resultados dos momentos resistentes e momentos actuantes das amostras simuladas pelo método do Hipercubo Latino

Método Hipercubo Latino						
N	$M_R(\text{kN.m})$		$M_S(\text{kN.m}) -100\text{kN/m}$		$M_S(\text{kN.m}) -130\text{kN/m}$	
	μ_R	σ_R	μ_S	σ_S	μ_S	σ_S
100	538.113	33.222	312.517	20.055	406.316	26.820
1000	538.192	32.633	312.527	19.874	406.310	26.235
5000	538.078	32.305	312.531	19.779	406.286	25.644
10000	538.092	32.583	312.524	19.817	406.285	25.849

Comparando os valores das médias do momento resultante e dos momentos actuantes, constata-se que não há uma variação acentuada entre as quatro simulações, sendo a maior delas, 3.6%, do desvio padrão do momento actuante, resultante de $q=130\text{kN/m}$. Assim pode-se afirmar que, mesmo com um número reduzido de simulações é possível obter valores já aproximados do que se obteria com simulações maiores, poupando esforço computacional. Estes dados vem confirmar o que já foi referido nesta dissertação, o método do Hipercubo Latino necessita de menos simulações para atingir a mesma precisão do método de Monte Carlo. Outro factor que faz com que o método do Hipercubo Latino seja mais rápido, é a ausência de testes de hipóteses na fase de simulação. Um utilizador que experimente os dois métodos de simulação no programa de apoio, vai reparar um decréscimo de tempo entre cada simulação de variáveis aleatórias. De seguida apresentam-se os valores obtidos para a resposta estrutural.

Tabela 3.16 - Resultados dos testes de hipóteses das amostras simuladas pelo método do Hipercubo Latino

Método Hipercubo Latino									
Teste de Hipótese	100		1000		5000		10000		
	Norm.	K-S	Norm.	K-S	Norm.	K-S	Norm.	K-S	
$M_R(\text{kN.m})$	a	-	a	-	d	0.745	r	0.760	
$M_S(\text{kN.m}) -100\text{Kn/m}$	a	-	a	-	a	-	d	1.040	
$M_S(\text{kN.m}) -130\text{Kn/m}$	a	-	a	-	a	-	a	-	

Após simular as variáveis aleatórias, o procedimento é igual nos dois métodos. Ou seja ao introduzir o momento resistente e os momentos actuantes, efectua-se os testes de hipótese para verificar a normalidade de cada uma das amostras. Analisando a tabela 3.16, verifica-se que só para as simulações 5000 e 10000 é que começa a haver desvios da normalidade. Contudo, todas as amostras verificam o teste de Kolmogorov-Smirnov.

Tabela 3.17 - Resposta estrutural para o método do Hipercubo Latino

Método Hipercubo Latino				
	Z – 100kN/m		Z – 130kN/m	
N	μ	σ	μ	σ
100	225.596	38.806	131.797	42.697
1000	225.665	38.209	131.882	41.871
5000	225.546	37.879	131.792	41.246
10000	225.568	38.136	131.806	41.591

À semelhança do momento resistente e dos momentos actuantes, as respostas estruturais (Tabela 3.17) têm uma variação de média e desvio padrão pequena (máximo 2.5%). Mais uma vez confirma-se que, para simulações pouco numerosas, o método de Hipercubo Latino devolve valores aproximados dos reais.

Tabela 3.18 - Resultados finais para o método do Hipercubo Latino

Método Hipercubo Latino								
	Para q=100kN/m				Para q=130kN/m			
N	β	Pf	COV	$\epsilon(\%)$	β	Pf	COV	$\epsilon(\%)$
100	5.813	3.06E-09	-	361443.4	3.087	0.001011	-	628.483
1000	5.906	1.75E-09	-	151131.3	3.062	0.000817	0.99959	221.152
5000	5.954	1.31E-09	-	78296.1	3.195	0.000698	0.99965	106.982
10000	5.914	1.66E-09	-	49070.2	3.169	0.000765	0.49981	72.3016

Analisando a tabela 3.18, constata-se outra vez a mesma situação das tabelas 3.15 e 3.17, os valores iniciais já são aproximados dos valores finais. Estudando os valores resultantes da amostragem de 10000 simulações, verifica-se que o erro é semelhante ao do método de Monte Carlo para o mesmo número de simulações.

Comparando os índices de fiabilidade, obtidos pelo método do Hipercubo Latino, com os valores mínimos recomendados no EC-0 (2002) expostos na tabela 2.2, constata-se que o resultado final deste método é semelhante ao do método anterior. Para $q=100kN/m$, verificam-se todas as classes de fiabilidade para os dois períodos de referência e para $q=130kN/m$, já não verifica a classe de fiabilidade RC1 para 50 anos de referência. As seguintes figuras contém histogramas com curvas teóricas ajustadas dos momentos resistentes e das respostas estruturais, para $q=130kN/m$, para amostragens de 100, 1000, 5000 e 10000 simulações.

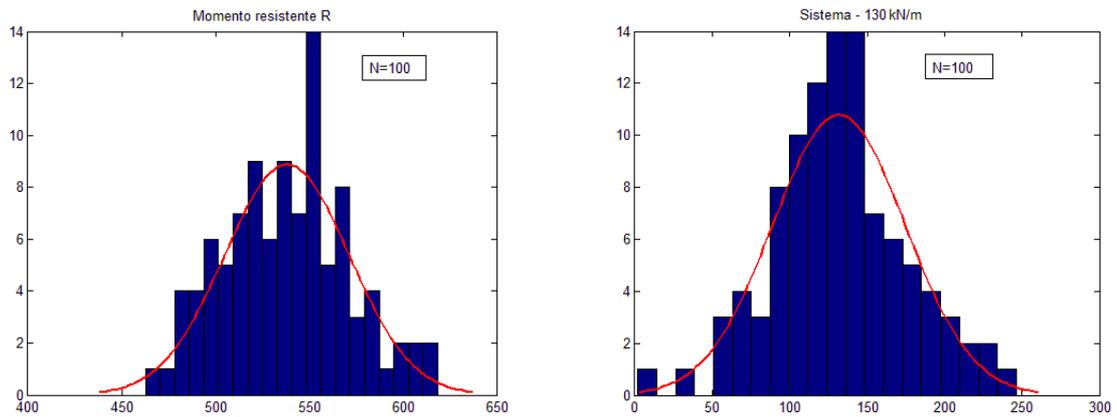


Fig. 3.17 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 100 amostras, pelo método do Hipercubo Latino

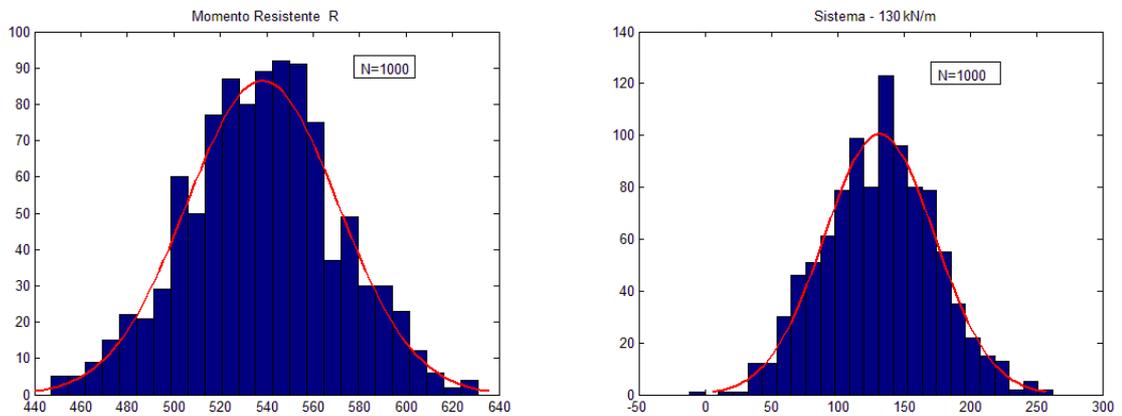


Fig. 3.18 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 1000 amostras, pelo método do Hipercubo Latino

Estudando o histograma da resposta estrutural na figura 3.18, verifica-se que já existem valores negativos na distribuição, daí o valor da COV. Recordando o mesmo histograma para o método de Monte Carlo, tal não acontecia. Mais uma vez confirma-se que para amostragens pequenas o método do Hipercubo Latino tem uma distribuição mais realista que o primeiro método.

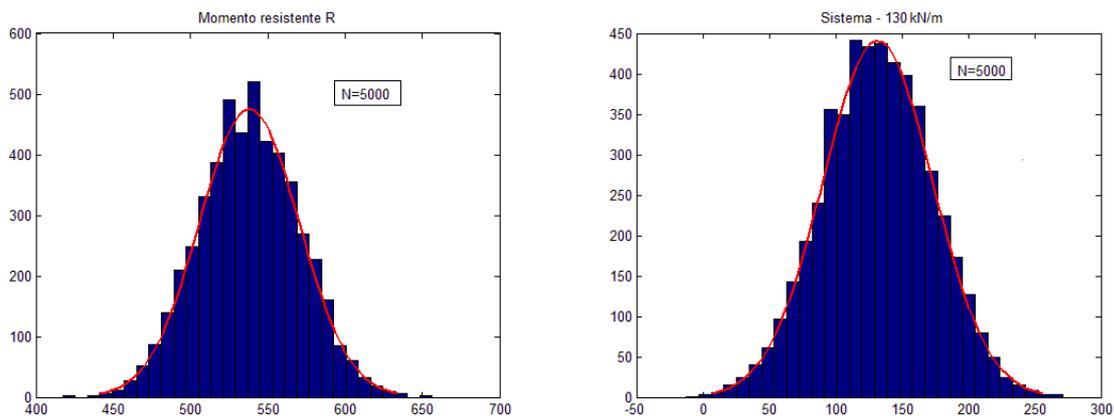


Fig. 3.19 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 5000 amostras, pelo método do Hiper cubo Latino

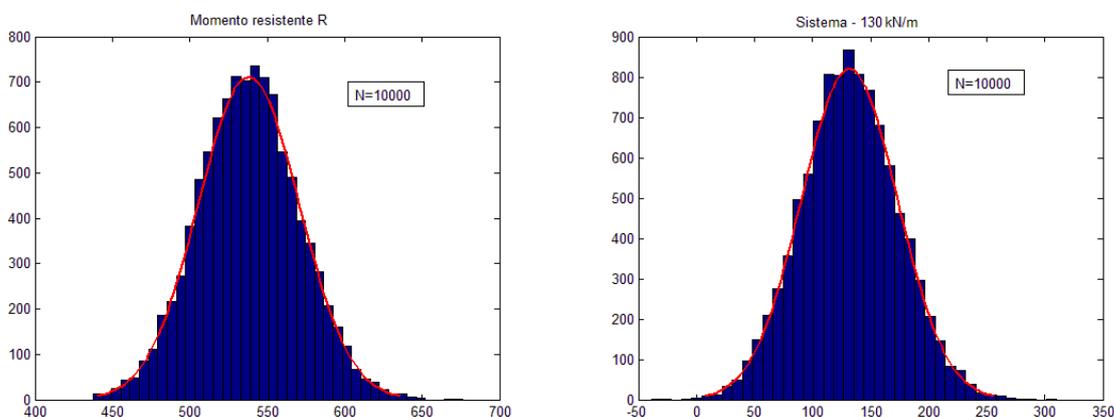


Fig. 3.20 - Momento resistente (à esquerda) e resposta estrutural (à direita) para 10000 amostras, pelo método do Hiper cubo Latino

Ao contrário dos histogramas de 100 e 1000 amostragens, os histogramas presentes nas figuras 3.19 e 3.20, já apresentam semelhança com os histogramas do método anterior, tal como os valores obtidos para o índice de fiabilidade e erro (%).

Dá-se por terminado o exemplo de aplicação e a introdução ao programa de apoio à avaliação de fiabilidade estrutural por métodos de simulação. Uma conclusão importante que se tira da análise efectuada, é o facto que, para o método do Hiper cubo Latino, são necessárias menos simulações para obter resultados com boa precisão, resultando em menor esforço computacional.

4

CASO PRÁTICO

4.1. INTRODUÇÃO

Pretende-se avaliar a fiabilidade estrutural de uma passagem superior recorrendo ao programa desenvolvido. A verificação de segurança estrutural será efectuada segundo o terceiro nível especificado na secção 2.2.4, ou seja via métodos probabilísticos. Assim este capítulo tem como objectivo principal calcular as medidas de fiabilidade estrutural, índice de fiabilidade e probabilidade de rotura, e verificar se os valores mínimos recomendados no EC-0 (2002), expostos na tabela 2.2, são respeitados. Depois de avaliar a passagem superior intacta considera-se uma possível deterioração das armaduras, ordinárias e de pré-esforço, e verifica-se se continuam a respeitar os valores mínimos recomendados.

É importante referir que o objectivo deste capítulo, é a aplicação do programa a um caso real e não o cálculo estrutural completo de uma passagem superior. Sendo assim serão feitas algumas simplificações e considerações, sempre sem prejudicar o objectivo deste capítulo.

Para atingir estes objectivos, todas as variáveis envolvidas serão simuladas pelos métodos de Monte Carlo e do Hipercubo Latino. Serão ainda, avaliados dois tipos secções, um solicitado pelo maior momento positivo (a meio vão), e outro solicitado pelo momento negativo máximo (sobre os pilares). Para obter estes valores máximos efectua-se duas combinações que serão apresentadas mais à frente. Dada a hiperestaticidade da estrutura, poderá ser efectuada uma redistribuição de esforços, se o índice de fiabilidade não verificar o valor mínimo recomendado.

4.2. GEOMETRIA, MATERIAIS E ACÇÕES

A passagem superior consiste numa estrutura simétrica em pórtico com extensão total de 40 metros. A passagem está dividida em 3 vãos, dois vãos extremos, cada um com 10 metros de extensão, e um vão central com 20 metros de extensão. O tabuleiro está apoiado em dois pilares e nos dois extremos. Existe continuidade estrutural entre os pilares e o tabuleiro

O pórtico é constituído a todo comprimento, por uma secção de betão armado em T como ilustra a figura 4.1. Existem duas soluções para armadura ordinária, uma para os momentos positivos máximos e outra para momentos negativos máximos, figura 4.2 e figura 4.3, respectivamente. Analisando as figuras percebe-se que falta armaduras construtivas e armaduras de resistência ao corte, mas como a intenção é calcular a resistência à flexão, ignoram-se essas armaduras. A secção definida pelas figuras 4.2 e 4.3, apresenta um banzo de 3 metros de largura e uma alma com 1.5 metros de largura. Na secção a meio vão, tem-se uma armadura inferior de 13Ø20 e a secção sobre o pilar tem-se uma armadura superior de 20Ø20.

Apesar de ilustrada uma única alma, esta pode ser considerada como uma alma equivalente. Ou seja, na realidade a secção pode ter duas almas com 0,75 metros de largura, mas para cálculo utiliza-se a largura equivalente.

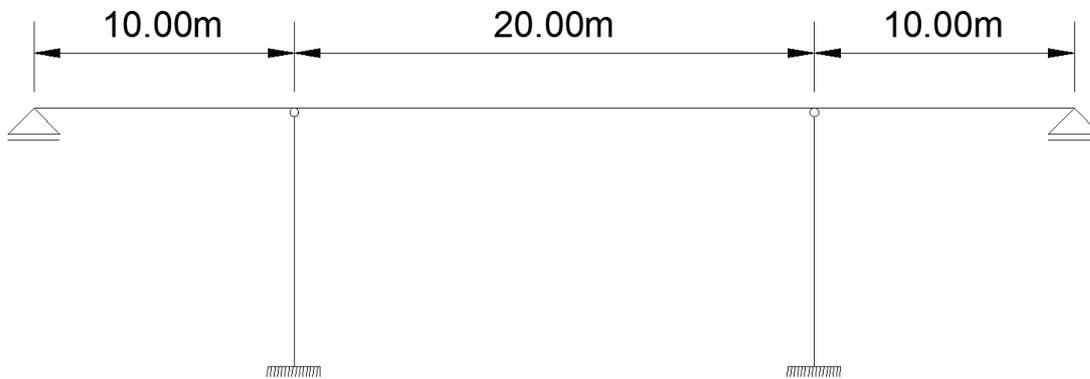


Fig. 4.1 - Modelo estrutural da passagem superior

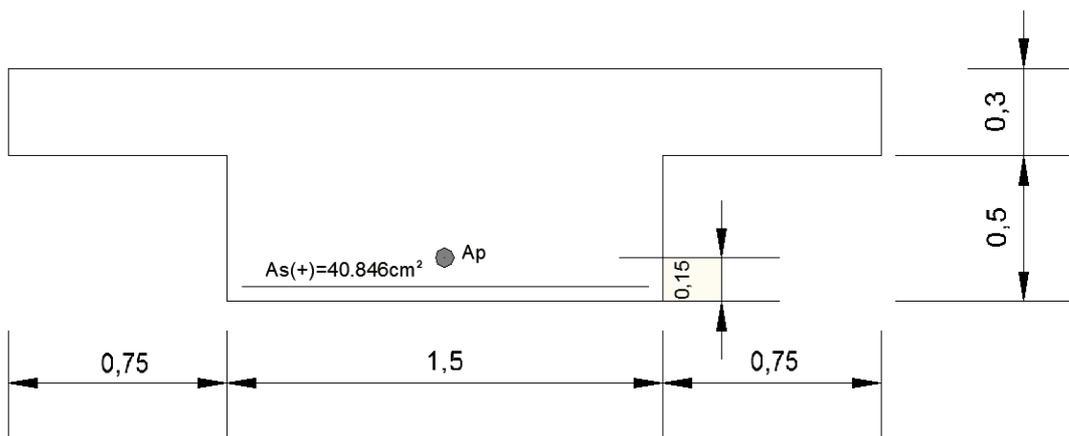


Fig. 4.2 - Secção a meio vão (aos 20 metros)

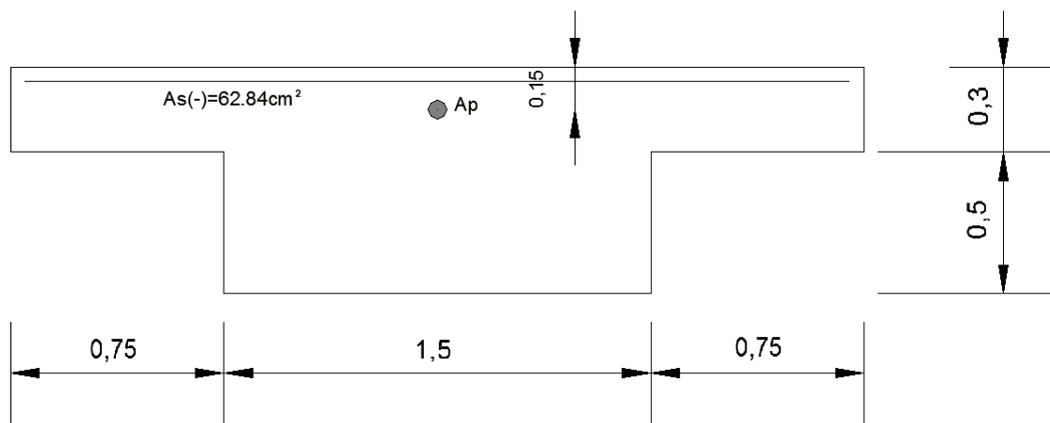


Fig. 4.3 - Secção onde se dá o encontro com os pilares (aos 10 e 30 metros)

Esta secção é ainda reforçada com uma solução de pré-esforço, usando dois cabos de cinco cordões de 0.5” que, para simplificação, consideram-se como um cabo com uma área total equivalente. O cabo de pré-esforço tem um traçado curvo, como ilustra a figura 4.4, sendo o ponto mais baixo, a meio vão, a 0,15 cm da face inferior da secção (figura 4.2) e o ponto mais alto, sobre o pilar, a 0,15 cm da face superior da secção (figura 4.3).

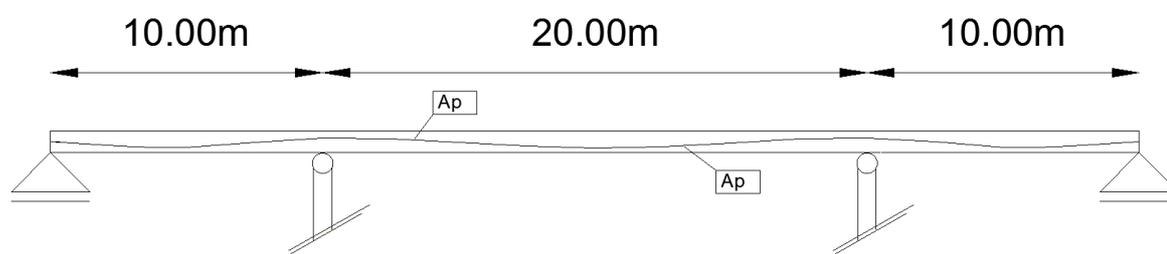


Fig. 4.4 - Traçado do cabo de pré-esforço equivalente

A passagem superior foi concebida com betão da classe $C25/30$, com aço ordinário da classe $S500$ e aço de pré-esforço da classe $S1680/1860$. É importante referir que, geralmente, para elementos pré-esforçados usa-se um betão igual ou superior a $C30/37$, mas como já se têm os valores médios para o $C25/30$. Os valores médios e respectivos desvios padrão encontram-se na tabela 4.2. Para obter estes valores médios e os seus coeficientes de variação, recorreu-se a dados de outros autores, mais concretamente Henriques (1998) e Wiśniewski (2007). Nas tabelas 4.1 e 4.2 estão referenciados todos os dados geométricos e materiais, com as respectivas médias e desvios padrão, que irão ser utilizados na avaliação de fiabilidade estrutural. É de notar que não vão ser incluídos dados referentes a módulos de elasticidade e extensões. Para simplificar os cálculos, considera-se que os aços ordinários e os de pré-esforço encontram-se no patamar de cedência. O coeficiente de variação entre a média e desvio padrão representa-se por “CV(%)”.

Tabela 4.1 - Parâmetros de distribuição das variáveis envolvidas (dados geométricos)

Variável Aleatória	Descrição	Tipo de distribuição	Média	CV(%)	Desvio padrão
Dados Geométricos					
B	- Largura do Banzo	Normal	300 cm	2	6 cm
b	- Largura da Alma	Normal	150 cm	2	3 cm
D	- Altura útil da armadura de pré-esforço	Normal	65 cm	2	1.3cm
d	- Altura útil da armadura ordinária	Normal	75 cm	2	1.5 cm
c	- Recobrimento	Normal	5 cm	2	0.1 cm

Tabela 4.2 - Parâmetros de distribuição das variáveis envolvidas (betão, aço ordinário e aço de pré-esforço)

Variável Aleatória	Descrição	Tipo de distribuição	Média	CV (%)	Desvio padrão
Betão C25/30					
f_c	- Tensão de rotura do betão C25/30	Normal	37.8 MPa	7.7	2.9106 MPa
Aço Ordinário S500					
f_{sy}	- Tensão de cedência da armadura (S500)	Normal	603 MPa	6	36.18 MPa
A_s	- Área da armadura	Normal	*	4.3	*
A_{si}	- Área da armadura inferior da secção a meio vão As nominal = 40.856 cm ²	Normal	37.578 cm ²	4.3	1.616 cm ²
A_{ss}	- Área da armadura superior da secção sobre o pilar As nominal = 62.84 cm ²	Normal	57.813 cm ²	4.3	2.486 cm ²
* - As real / As nominal = 0.92					
Aço de Pré-Esforço S1680/1860 – 0.5"					
f_{pu}	- Tensão de rotura	Normal	1902 MPa	2	38.04 MPa
$f_{p0.1}$	- Tensão limite de proporcionalidade a 0.1% $f_{p0.1} = f_{pu} * 0.9$	Normal	1711.8 MPa	2	34.23 MPa
A_p	- Armadura de pré-esforço (x10)	Normal	0.93 cm ²	2	0.0186 cm ²

A passagem superior está sujeita a dois tipos de acções, permanentes e sobrecargas. As primeiras são, o peso próprio da estrutura, o peso dos revestimentos e dos acessórios necessários, como o guarda corpos e sistemas de drenagem de águas. Na tabela 4.3 estão expostos os valores de cada acção em kN/m.

Tabela 4.3 - Acções actuantes na passagem superior

	PP – peso próprio	Prev – peso dos revestimentos e acessórios	Psc – peso da sobrecarga
Média (kN/m)	41.25	10	20
CV(%)	8	10	15
Desvio Padrão (kN/m)	3.3	1	3

Tendo em atenção as características de cada carga, estabelece-se que as cargas permanentes estão distribuídas uniformemente a todo o comprimento da passagem superior. Para a sobrecarga é necessário combinar as acções e determinar, quais são as combinações que provocam maior momento negativo e positivo. Como se trata de uma passagem dividida em três partes, é fácil determinar estas duas combinações. A combinação 1 provoca momentos máximos positivos (Fig. 4.5) e a combinação 2 provoca os momentos máximos negativos sobre o pilar da esquerda (Fig. 4.6).

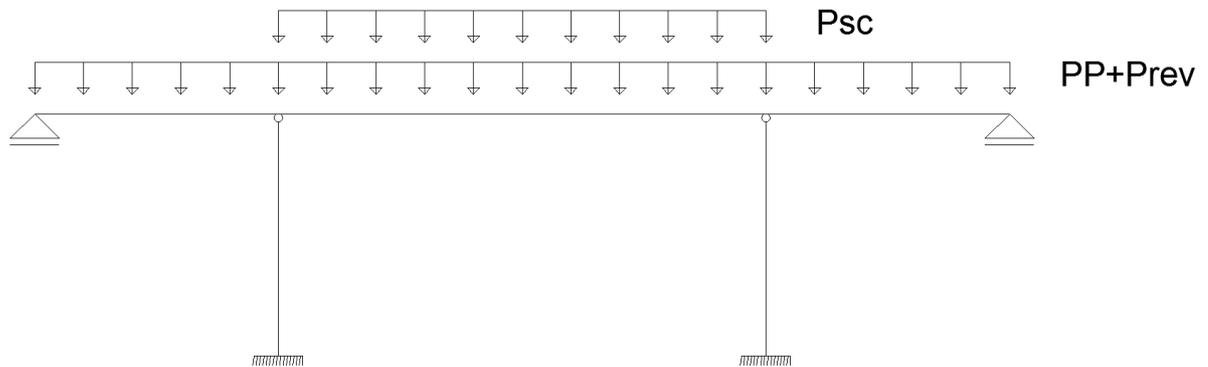


Fig. 4.5 - Esquema de cargas aplicadas para a combinação 1

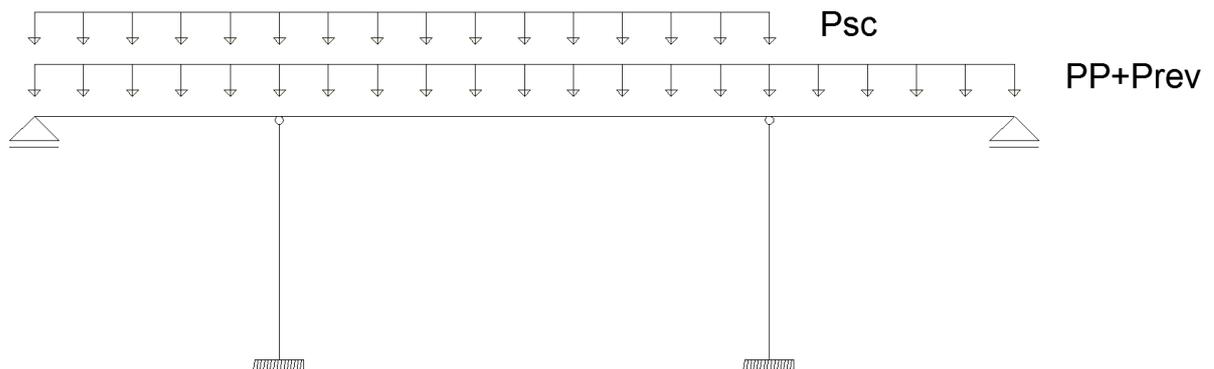


Fig. 4.6 - Esquema de cargas aplicadas para a combinação 2

Na tabela 4.4 estão expostos os momentos resultantes das cargas nas duas combinações consideradas. M_{pp} corresponde ao momento actuante proveniente do peso próprio do tabuleiro, M_{rev} ao momento actuante provenientes dos revestimentos e acessórios e M_{sc1} e M_{sc2} aos momentos actuantes provenientes da sobrecarga na combinação 1 e 2, respectivamente.

Tabela 4.4 - Momentos actuantes na passagem superior

Momentos actuantes	Mpp	Mrev	Msc1	Msc2	
Máximo positivo (kN.m)	902.3	218.8	500	469.7	
Máximo negativo (kN.m)	1160.2	281.3	500	593.8 (esquerda)	468.8 (direita)
Tipo de distribuição	Normal	Normal	Normal	Normal	
CV(%)	8	10	15	15	
Desvio Padrão positivo (kN.m)	72.184	21.88	75	70.455	
Desvio Padrão negativo (kN.m)	92.816	28.13	75	89.07	70.32

4.3. AVALIAÇÃO DE FIABILIDADE ESTRUTURAL

Com as variáveis aleatórias definidas e caracterizadas estatisticamente, procede-se à avaliação da fiabilidade estrutural da passagem superior. Começa-se por simular 10000 amostras pelo método de Monte Carlo, se o erro não for admissível aumenta-se o número de amostras. Se após aumentar o número de amostras os valores médios e desvios padrão da resposta estrutural forem semelhantes, considera-se que a simulação tem precisão suficiente. Com o intuito de comparar os resultados do método de Monte Carlo, procede-se também à simulação de 100 amostras pelo método do Hipercubo Latino.

Como já foi referido, avalia-se a fiabilidade estrutural para duas secções da passagem superior, uma onde o momento positivo máximo actua e outra onde o momento negativo máximo actua. O cálculo do momento resistente de cada secção, será efectuado a partir das equações de equilíbrio de forças internas da secção, recorrendo ao programa Excel, tal como no exemplo de aplicação da secção 3.4. Os resultados obtidos no programa Excel, serão importados para o programa de apoio, onde será efectuado o cálculo do índice de fiabilidade, da probabilidade de rotura e ainda do erro.

Sendo assim inicia-se a simulação de 10000 amostras pelo método de Monte Carlo, introduzindo todas as variáveis aleatórias no programa de apoio, devidamente caracterizadas.

4.3.1. SIMULAÇÃO DE 10000 AMOSTRAS PELO MÉTODO DE MONTE CARLO

Tabela 4.5 - Resultados dos testes de Hipótese das variáveis simuladas (Monte Carlo, 10000 amostras)

	B	b	D	d	c	f_c	f_{sy}	A_{si}	A_{ss}
Teste de Normalidade	a	a	d	a	d	a	a	d	d
Teste K-S	-1	-1	0,561	-1	1,133	-1	-1	1,082	0,906
	$f_{p0.1}$	Ap	Mpp+	Mrev+	Msc+	Mpp-	Mrev-	Msc-	
Teste de Normalidade	d	a	a	a	a	a	a	d	
Teste K-S	0,563	-1	-1	-1	-1	-1	-1	0,676	

Na tabela 4.5 encontram-se os resultados dos testes de hipótese, de normalidade e de Kolmogorov-Smirnov. As últimas seis variáveis têm um valor positivo ou negativo à direita para distinguir entre a secção a meio vão (+) e a secção sobre o pilar (-).

Os resultados do teste de normalidade são interpretados do seguinte modo:

- “a” - verifica as duas condições deste teste.
- “d” - só verifica a segunda condição e será efectuada uma verificação do teste de Kolmogorov-Smirnov.
- “n” - as amostras foram simuladas pelo método do Hipercubo Latino ou não têm uma distribuição normal, sendo assim não necessitam de verificar este teste.
- “r” – não verifica a hipótese de normalidade.

Os resultados do teste de Kolmogorov-Smirnov são interpretados do seguinte modo:

- “-1” - este teste não foi efectuado, ou porque verificou o teste de normalidade, ou porque as amostras foram simuladas pelo método do Hipercubo latino.
- [0,1.364] – as amostras verificam este teste.
- “-2” – as amostras não verificam este teste.

Nota: Se um conjunto de amostras não verificar os testes de hipótese no primeiro módulo, o programa recusa este conjunto e repete a simulação. Assim os resultados “r” (teste de normalidade) e “-2” (teste Kolmogorov-Smirnov) só se aplicam ao segundo módulo do programa de apoio, quando se verifica a normalidade dos momentos resistentes e dos momentos actuantes.

Na tabela 4.6 estão expostos os valores médios, desvios padrão, valores máximos e mínimos dos momentos resistentes e actuantes. Estes valores são obtidos através de cálculos efectuados numa folha Excel. Comparando os momentos resistentes com os actuantes, conclui-se que para esta amostragem não haverá um único caso de rotura, ou seja o COV vai ser indefinido. Dada a diferença entre os valores mínimos dos momentos resistentes e os valores máximos dos actuantes, adivinha-se que o índice de fiabilidade será grande e por consequência, a probabilidade de rotura, pequena. De seguida são importados para o programa de apoio, onde o primeiro procedimento é a verificação de normalidade.

Tabela 4.6 – Média, desvio padrão, mínimos, máximos e resultados dos testes de Hipótese dos momentos resistentes e actuantes (Monte Carlo, 10000 amostras)

N=10000	Mr+	Mr-	Ms+	Ms-
Média (kN.m)	2667.46	3420.24	1620.85	2035.78
Desvio Padrão (kN.m)	128.24	180.32	105.49	130.77
Mínimo (kN.m)	2206.68	2737.17	1257.64	1501.37
Máximo (kN.m)	3149.26	4123.85	1981.35	2540.40
Teste de normalidade	r	r	a	R
Teste de K-S	0.790	1.141	-1	1.047

Analisando os dados da tabela 4.7, constata-se que a média da resposta estrutural é muito elevada, o que significa que os índices de fiabilidade serão elevados. Isto significa, que existe uma grande

margem entre o momento resistente e o momento actuante para ambas as secções. Estudando os valores mínimos e máximos, confirma-se esse facto.

Tabela 4.7 – Média, desvio padrão, mínimo e máximo da resposta estrutural (Monte Carlo, 10000 amostras)

N=10000	Z- Média (kN.m)	Z- Desvio Padrão (kN.m)	CV (%)	Mínimo (kN.m)	Máximo (kN.m)
Secção a meio vão	1046.61	166.05	15.86	389.11	1780.37
Secção sobre o pilar	1384,46	222,75	16.09	588.14	2149.35

Na tabela 4.8 estão expostos os resultados finais da simulação de 10000 amostras pelo método de Monte Carlo. Verifica-se o previsto, índices de fiabilidade muito grandes, com probabilidades de rotura muito pequena e erros significativos. Recordando a fórmula do erro (2.100) da secção 2.3.2.3, quanto menor for a probabilidade de rotura e menor o numero de amostras, maior vai ser o erro. Analisando o erro existente para as duas secções, conclui-se que, para atingir um erro na ordem dos 10%, seriam necessárias aproximadamente $2.8E12$ e $1.6E12$ amostras na secção a meio vão e a secção sobre o pilar, respectivamente. Como a probabilidade de rotura e o erro são inversamente proporcionais, é usual que o erro seja tão grande, contudo estes valores são meramente indicativos. É óbvio que para obter uma simulação precisa, não é necessário simular um conjunto de amostras com esta dimensão. Se ao aumentar o número de amostras simuladas, os resultados se mantiverem aproximados, considera-se que a simulação tem precisão suficiente. Então irá simular-se para 100000 amostras e verificar se há grandes alterações nos resultados obtidos. De seguida apresentam-se os histogramas com curvas teóricas ajustadas para os momentos resistentes e actuantes e ainda para a resposta estrutural.

Tabela 4.8 – Resultados finais da avaliação de fiabilidade estrutural (Monte Carlo, 10000 amostras)

N=10000	P_F	β	COV	Erro (%)
Secção a meio vão	1.46E-10	6.303	-	165486
Secção sobre o pilar	2.56E-10	6.215	-	124973

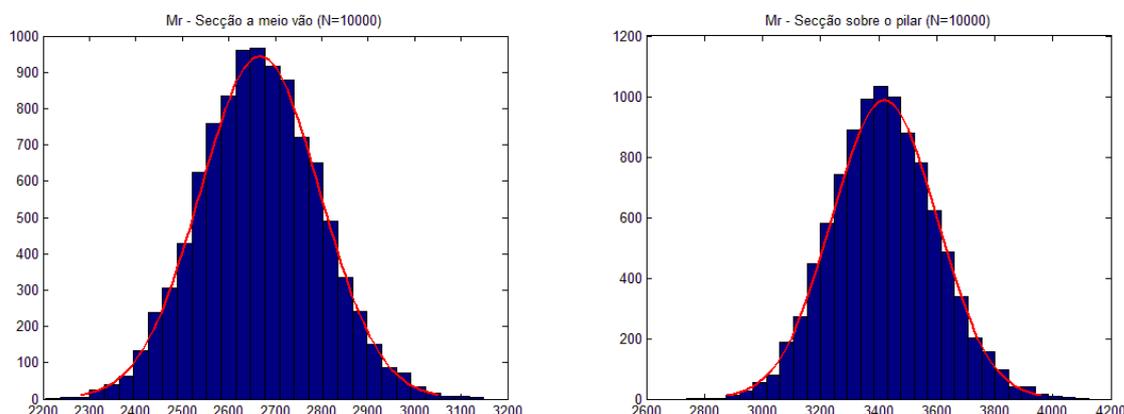


Fig. 4.7 - Histogramas com curva teórica ajustada dos momentos resistentes para 10000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

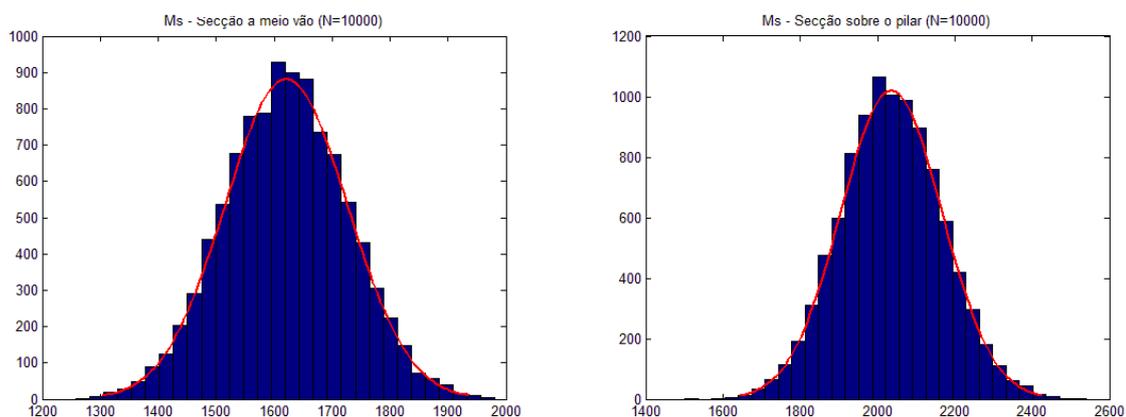


Fig. 4.8 - Histogramas com curva teórica ajustada dos momentos actuantes para 10000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

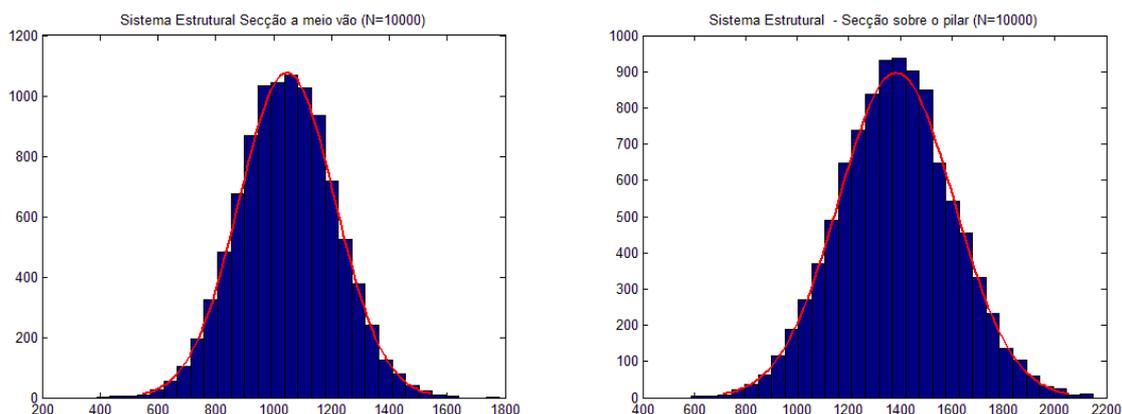


Fig. 4.9 - Histogramas com curva teórica ajustada das respostas estruturais para 10000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

Analisando a figura 4.9, verifica-se que para a simulação de 10000 amostras não ocorre uma única vez um caso de rotura. Estes histogramas vêm confirmar a comparação dos momentos mínimos resistentes e máximos actuantes (Tabela 4.6), pois estes nunca se chegam a cruzar. Para a secção a meio vão o ponto mais baixo é próximo dos $400kN.m$ e para a secção sobre o pilar é próximo dos $600kN.m$. Este facto vem confirmar a ausência de valor de COV na tabela 4.8. Este coeficiente só pode ser calculado se em todo espaço amostral existirem casos de rotura. Proceder-se então à simulação de 100000 amostras pelo método de Monte Carlo.

4.3.2. SIMULAÇÃO DE 100000 AMOSTRAS PELO MÉTODO DE MONTE CARLO

Mais uma vez, começa-se por introduzir as variáveis aleatórias, devidamente caracterizadas, no programa de apoio à avaliação de fiabilidade. Na tabela 4.9, encontram-se os resultados dos testes de hipótese efectuados imediatamente após a simulação.

Tabela 4.9 - Resultados dos testes de Hipótese das variáveis simuladas (Monte Carlo, 100000 amostras)

	B	b	D	d	c	f _c	f _{sy}	A _{si}	A _{ss}
Teste de Normalidade	a	a	a	d	a	a	d	a	a
Teste K-S	-1	-1	-1	0,683	-1	-1	0,779	-1	-1
	f _{p0.1}	Ap	Mpp+	Mrev+	Msc+	Mpp-	Mrev-	Msc-	
Teste de Normalidade	a	a	d	a	d	a	d	a	
Teste K-S	-1	-1	0,910	-1	0,814	-1	1,050	-1	

Com os valores exportados para Excel, procede-se ao cálculo dos momentos resistentes e actuantes, tal como na simulação anterior. As médias, os desvios padrão, valores mínimos e máximos destes momentos estão expostos na tabela 4.10. Recordando os valores da tabela 4.6, verifica-se que de 10000 amostras para 100000 amostras não houve grande variação nos valores médios e nos desvios padrão. Isto significa, que apesar de existir um erro grande, os valores da simulação de 10000 amostras já são aproximados. Nos valores mínimos e máximos já ocorre uma variação considerável. Para esta simulação, obtêm-se valores extremos que não estavam presentes na simulação anterior, ou seja um conjunto de amostras com maior amplitude. Confrontando os momentos resistentes com os actuantes, chega-se à mesma conclusão da simulação anterior, não há casos de rotura (COV indefinido).

Ainda na tabela 4.10, encontram-se os resultados dos testes de hipótese do segundo módulo. Analisando os dados dos testes das duas simulações, reconhece-se que os dois momentos resistentes não verificam o teste de normalidade e também não verificam o teste de Kolmogorov-Smirnov, ao contrário do sucedido na primeira simulação.

Tabela 4.10 - Média, desvio padrão, mínimos, máximos e resultados dos testes de Hipótese dos momentos resistentes e actuantes (Monte Carlo, 100000 amostras)

N=100000	Mr+	Mr-	Ms+	Ms-
Média (kN.m)	2668.34	3420.62	1621.38	2035.33
Desvio Padrão (kN.m)	129.23	181.77	106.46	131.86
Mínimo (kN.m)	2189.16	2729.29	1141.76	1428.98
Máximo (kN.m)	3312.28	4252.65	2074.46	2621.80
Teste de normalidade	r	r	a	a
Teste de K-S	-2	-2	-1	-1

Na tabela 4.11, estão expostos os valores médios, os desvios padrão, os mínimos e os máximos da resposta estrutural das duas secções em análise. Como é possível constatar, as médias e desvios obtidos nesta simulação são muito semelhantes à anterior. Mais uma vez, confirma-se o facto de a simulação de 10000 amostras, já ser suficientemente aproximada. Tal como sucedeu com os momentos resistentes e actuantes, os valores mínimos e máximos da resposta estrutural apresentam

valores extremos que não estão presentes na simulação anterior. Segue-se o cálculo do índice de fiabilidade e da probabilidade de rotura.

Tabela 4.11 - Média, desvio padrão, mínimo e máximo da resposta estrutural (Monte Carlo, 100000 amostras)

N=100000	Z- Média (kN.m)	Z- Desvio Padrão (kN.m)	CV (%)	Mínimo (kN.m)	Máximo (kN.m)
Secção a meio vão	1046.96	167.43	15.99	313.90	1752.07
Secção sobre o pilar	1385.29	224.56	16.21	506.00	2447.59

Os resultados finais desta avaliação de fiabilidade estrutural encontram-se na tabela 4.12. Recordando os valores obtidos para a simulação de 10000 amostras, constata-se que o resultado final é ligeiramente diferente. O índice de fiabilidade diminui, levando ao aumento da probabilidade de rotura. Contudo a probabilidade continua na ordem dos 10^{-10} , o que significa que mesmo aumentando o número de simulações este valor não irá alterar muito. Comparando os valores dos erros obtidos nesta simulação com os anteriores, constata-se que houve um decréscimo considerável do erro. Todavia utilizando a equação 2.98 para calcular o número de simulações necessárias para obter um erro na ordem dos 10%, chega-se à mesma conclusão, para obter um erro na ordem dos 10% são necessárias simulações de um número de amostras excessivo. É importante referir que o tempo de computação necessário para efectuar amostras com esta grandeza, já é considerável. Tendo em conta que se tem 17 variáveis aleatórias envolvidas e que cada é caracterizada por 100000 amostras, é usual que o tempo de computação seja excessivo. Pode então concluir-se que, aumentar o número de amostras resulta dados mais precisos. Contudo, simulações na ordem das 10000 amostras já apresentam precisão admissível. De seguida apresentam se os histogramas dos momentos resistentes e actuantes e a resposta estrutural das duas secções.

Tabela 4.12 - Resultados finais da avaliação de fiabilidade estrutural (Monte Carlo, 100000 amostras)

N=100000	P_F	β	COV	Erro (%)
Secção a meio vão	2.01E-10	6.253	-	44576
Secção sobre o pilar	3.44E-10	6.169	-	34109

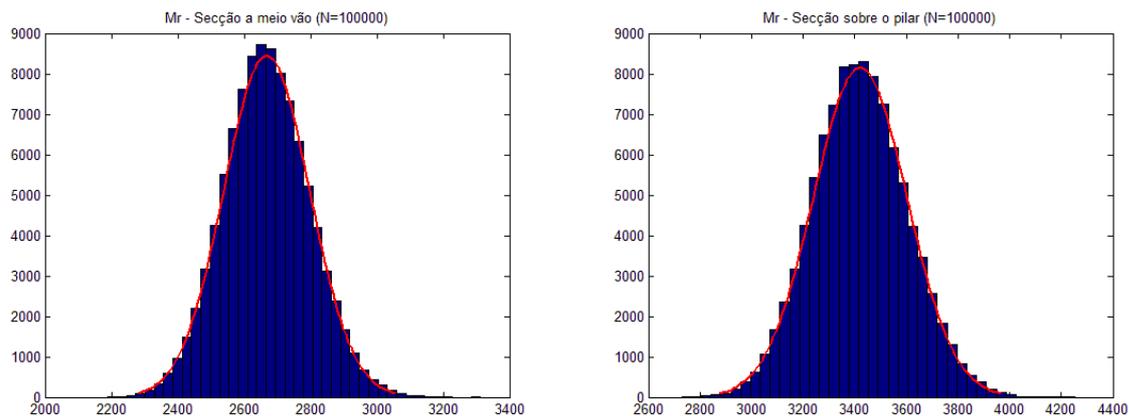


Fig. 4.10 - Histogramas com curva teórica ajustada dos momentos resistentes para 100000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

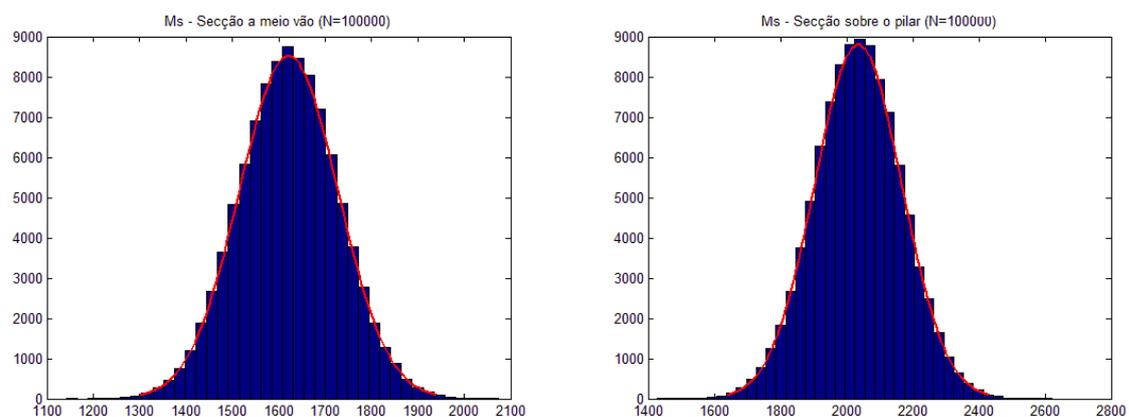


Fig. 4.11 - Histogramas com curva teórica ajustada dos momentos actuantes para 100000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

Observando as figuras 4.10 e 4.11, é possível distinguir uma pequena diferença com os histogramas anteriores. As caudas das distribuições de probabilidade são ligeiramente mais alongadas, atingindo valores mais extremos (mínimos e máximos), que a simulação de 10000 amostras não atinge. Estes histogramas confirmam os dados obtidos da tabela 4.11, onde se comentou que esta simulação obtém valores extremos que não estavam presentes na simulação anterior.

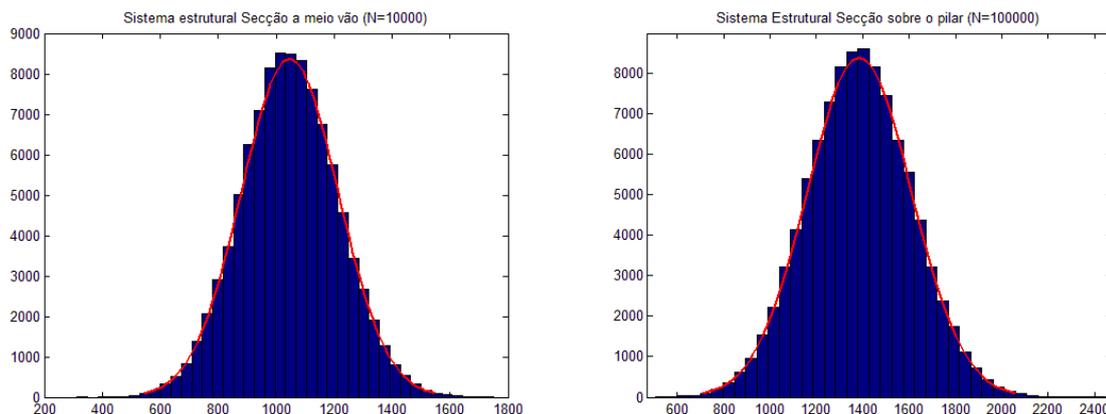


Fig. 4.12 - Histogramas com curva teórica ajustada das respostas estruturais para 100000 amostras pelo método de Monte Carlo (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

Comparando a resposta estrutural das duas simulações, constata-se uns valores mínimos e máximos mais extremos, tal como aconteceu com os momentos resistentes e actuantes. Mesmo com estes momentos mais alargados não ocorre um único caso de rotura em todo o espaço amostral, daí o valor indefinido do COV da tabela 4.12. Segue-se a simulação de 100 amostras pelo método do Hipercubo Latino.

4.3.3. SIMULAÇÃO DE 100 AMOSTRAS PELO MÉTODO DO HIPERCUBO LATINO

Com esta simulação, pretende-se comparar a precisão do método do Hipercubo latino com o anterior e ainda, o tempo de computação necessário para cada método. A metodologia usada para simular as variáveis aleatórias encontra-se descrita na secção 2.3.3.1. É de notar que, para este método não serão efectuados testes de hipótese no primeiro módulo, pois ao utilizar este método garante-se um comportamento segundo a lei teórica.

Comparando as médias e desvios padrão dos momentos resistentes e actuantes de cada secção (Tabela 4.13), com os valores das duas simulações anteriores, verifica-se que este método apresenta valores muito aproximados. Tal como no exemplo de aplicação do capítulo 3, este método é capaz de apresentar valores precisos com um número de amostras reduzidas. Todavia os valores mínimos e máximos dos momentos, já apresentam uma variação considerável. Isto deve-se à própria metodologia do Hipercubo Latino. Este método só iria apresentar a mesma gama de valores se aumentasse o número de amostras, enquanto que para o método de Monte Carlo é totalmente aleatório.

No segundo módulo, os dados provenientes dos dois métodos de simulação, são tratados do mesmo modo. Ou seja os momentos são sujeitos a testes de hipótese. Estudando os resultados expostos na tabela 4.13, verifica-se que todos os momentos verificam o teste da normalidade.

Tabela 4.13 - Média, desvio padrão, mínimos, máximos e resultados dos testes de Hipótese dos momentos resistentes e actuantes (Hiper cubo Latino, 100 amostras)

N=100	Mr+	Mr-	Ms+	Ms-
Média (kN.m)	2667.82	3419.56	1621.10	2035.30
Desvio Padrão (kN.m)	132.78	180.20	108.53	133.04
Mínimo (kN.m)	2353.81	3018.80	1366.10	1680.31
Máximo (kN.m)	2955.17	3836.81	1954.24	2406.27
Teste de normalidade	a	a	a	a
Teste de K-S	-1	-1	-1	-1

Na tabela 4.14, estão expostos os valores médios, os desvios padrão, os mínimos e os máximos da resposta estrutural das duas secções em análise. Como é possível observar, as médias das respostas estruturais são muito semelhantes às obtidas nas simulações anteriores. Os desvios padrão já apresentam uma diferença ligeiramente superior, mas mais uma vez a grande diferença encontra-se nos valores mínimos e máximos. Tendo em conta que se está a comparar uma simulação de 100 amostras com duas de 10000 e de 100000 amostras, pode-se concluir que, até para amostras pequenas, o método do Hiper cubo Latino é muito preciso.

Tabela 4.14 - Média, desvio padrão, mínimo e máximo da resposta estrutural (Hiper cubo Latino, 100 amostras)

N=100	Z- Média (kN.m)	Z- Desvio Padrão (kN.m)	CV (%)	Mínimo (kN.m)	Máximo (kN.m)
Secção a meio vão	1046.72	171.49	16.38	641.79	1460.92
Secção sobre o pilar	1384.26	223.98	16.18	836.97	1904.76

Estudando os dados obtidos da simulação de 100 amostras pelo método do Hiper cubo latino (Tabela 4.15), constata-se que os valores finais são semelhantes aos valores obtidos pelas simulações anteriores. Na secção sobre o pilar, esta simulação é mais aproximada à simulação das 100000 amostras que a de 10000. Na secção a meio vão, o valor do índice de fiabilidade já é ligeiramente afastado, mas mantendo um nível de aproximação aceitável. Como seria de esperar o erro é muito elevado. Com uma probabilidade de rotura tão baixa e um número de amostras também baixo é usual que o erro seja ainda maior que as simulações anteriores. Ou seja, tal como nas simulações anteriores, para obter um erro admissível seria necessário simular muitas amostras. Sabendo que os valores mínimos da resposta estrutural não são inferiores a zero, é normal obter COV indefinidos e de esperar que os histogramas da resposta estrutural não contenham frequências para valores negativos.

Tabela 4.15 - Resultados finais da avaliação de fiabilidade estrutural (Hiper cubo Latino, 100 amostras)

N=100	P_F	β	COV	Erro (%)
Secção a meio vão	5.19E-10	6.104	-	878154
Secção sobre o pilar	3.2E-10	6.180	-	1117671

Comparativamente ao método de Monte Carlo, o tempo de computação no método do Hiper cubo Latino é muito inferior. Para além de simular menos amostras para obter resultados precisos, não necessita de efectuar testes de hipótese. Estes dois factos, fazem com que o método do Hiper cubo Latino seja mais eficiente. De seguida apresentam-se os histogramas dos momentos resistentes e actuantes e ainda a resposta estrutural das duas secções.

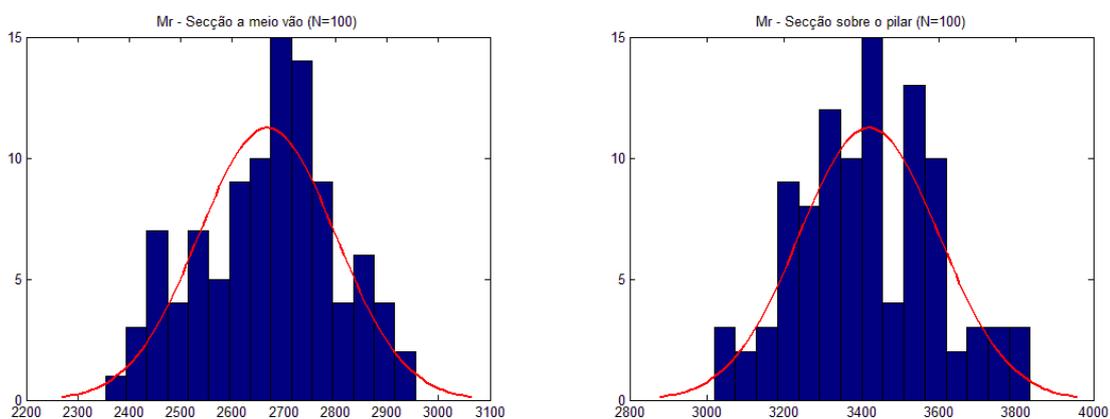


Fig. 4.13 - Histogramas com curva teórica ajustada dos momentos resistentes para 100 amostras pelo método de Hiper cubo Latino (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

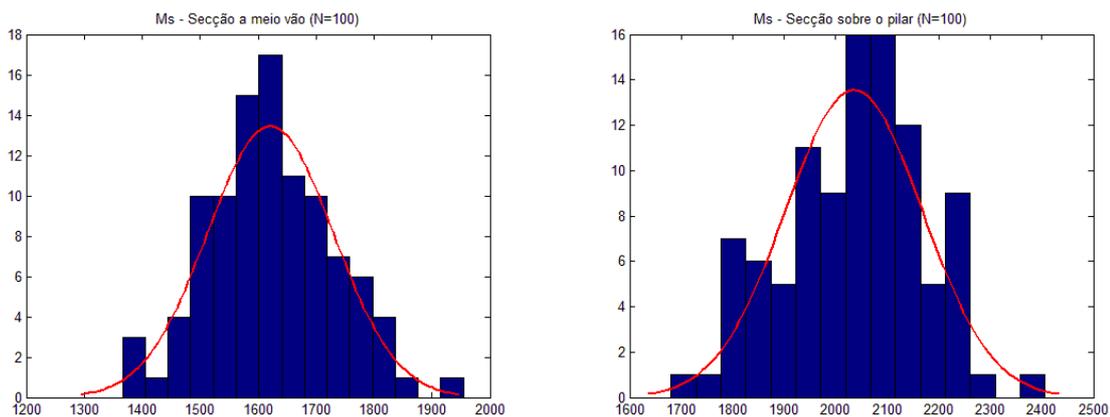


Fig. 4.14 - Histogramas com curva teórica ajustada dos momentos actuantes para 100 amostras pelo método de Hiper cubo Latino (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

Como seria de esperar, analisando os histogramas dos momentos resistentes e actuantes ilustrados nas figuras 4.13 e 4.14, verifica-se que o intervalo entre os valores mínimos e máximos é muito inferior

aos intervalos obtidos nas simulações anteriores. Na figura 4.15 estão ilustrados os histogramas para as respostas estruturais. Observando estes histogramas, confirmam-se todas as conclusões que foram tiradas das tabelas e histogramas anteriores.

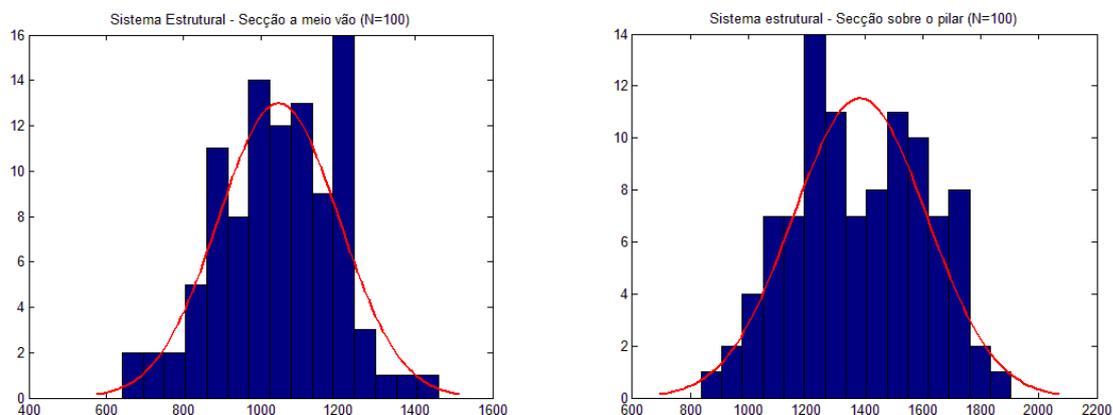


Fig. 4.15 - Histogramas com curva teórica ajustada das respostas estruturais para 100 amostras pelo método de Hipercubo Latino (à esquerda para a secção a meio vão e à direita para a secção sobre o pilar)

4.3.4. COMENTÁRIOS FINAIS

Antes de comparar as simulações efectuadas é importante referir que não foram simuladas 100 amostras através do método de Monte Carlo, porque para amostras reduzidas este método é muito impreciso, apresentando resultados pouco satisfatórios. Este facto já foi comprovado no exemplo de aplicação do terceiro capítulo.

Para comparar as simulações efectuadas, têm-se em conta três parâmetros, precisão, campo amostral e eficiência. Estes parâmetros podem ser caracterizados pelas médias, desvios padrão, mínimos e máximos da resposta estrutural, índices de fiabilidade e ainda tempos de computação. É importante referir que se considera que a simulação de 100000 amostras é a mais precisa, sendo por isso a referência de precisão. Começa-se por comparar a precisão das três simulações (Tabela 4.16):

- Média da resposta estrutural: Como tem vindo a ser comentado nas três simulações efectuadas, as médias das respostas estruturais são todas muito semelhantes, sendo a variação entre as mesmas, inferior a 0.1%. Isto significa que, para as médias, todas as simulações apresentam um elevado grau de precisão.
- Desvio padrão da resposta estrutural: Para comparar os desvios padrão das três simulações, compara-se variação do desvio em relação ao desvio da simulação de 100000 amostras. Estudando as variações presentes na tabela 4.17, constata-se que a primeira simulação apresenta duas variações semelhantes (0.8%) e que a terceira simulação já apresenta variações muito distintas (2,5% e 0.3%). Considerando estes dados, conclui-se que a primeira simulação é mais precisa no cálculo de desvios padrão.
- Índices de fiabilidade e probabilidade de rotura (Tabela 4.18): Analisando os dados obtidos, verifica-se que na secção sobre o pilar, as três simulações apresentam valores muito aproximados. Na secção a meio vão a terceira simulação já apresenta valores ligeiramente afastados. Este afastamento das outras simulações é, provavelmente, provocado pela diferença do desvio padrão. Todavia continua a ser suficientemente aproximado.

De seguida analisa-se a diferença entre os campos amostrais. Um campo amostral representa a gama de valores de uma simulação. Ou seja um campo amostral grande representa um intervalo grande entre o valor mínimo e o valor máximo. Analisando a tabela 4.16, facilmente se distingue a diferença entre o método Monte Carlo e o método do Hipercubo Latino. Se a intenção da simulação é obter uma gama de valores variada, recomenda-se simulações com um número elevado de amostras, independentemente do método.

Por último compara-se a eficiência. Considera-se uma simulação eficiente se usar poucos recursos e produzir resultados com precisão admissível. Sendo que, por recursos entendem-se, número de amostras e tempo de computação necessário. Tendo em conta as conclusões tiradas nos pontos anteriores, número reduzido de amostras usadas pelo método do Hipercubo Latino e ainda o tempo de computação reduzido do mesmo método, pode-se concluir que o método do Hipercubo Latino é sem dúvida o método mais eficiente.

Tabela 4.16 - Resumo das respostas estruturais obtidas nas simulações efectuadas

	Z- Média (kN.m)	Z- Desvio Padrão (kN.m)	CV (%)	Mínimo (kN.m)	Máximo (kN.m)	Intervalo (kN.m)
Monte Carlo N=10000						
Secção a meio vão	1046.61	166.05	15.86	389.11	1780.37	1391.26
Secção sobre o pilar	1384.46	222.75	16.09	588.14	2149.35	1561.22
Monte Carlo N=100000						
Secção a meio vão	1046.96	167.43	15.99	313.90	1752.07	1438.17
Secção sobre o pilar	1385.29	224.56	16.21	506.00	2447.59	1941.58
Hipercubo Latino N=100						
Secção a meio vão	1046.72	171.49	16.38	641.79	1460.92	819.14
Secção sobre o pilar	1384.26	223.98	16.18	836.97	1904.76	1067.79

Tabela 4.17 - Variação do desvio padrão, usando como referência a simulação pelo método de Monte Carlo de 100000 amostras

	Secção a meio vão	Variação em relação ao desvio padrão da segunda simulação (MC -N=100000)	Secção sobre o pilar	Variação em relação ao desvio padrão da segunda simulação (MC -N=100000)
Monte Carlo N=10000				
Desvio padrão de Z	166.05	0.824%	222,75	0.806%
Hipercubo Latino N=100				
Desvio padrão de Z	171.49	2.42%	223.98	0.256%

Tabela 4.18 - Resumo dos índices de fiabilidade e da probabilidade de rotura obtidos para todas as simulações

	P_F	β	Erro (%)
Monte Carlo N=10000			
Secção a meio vão	1.46E-10	6.303	165486
Secção sobre o pilar	2.56E-10	6.215	124973
Monte Carlo N=100000			
Secção a meio vão	2.01E-10	6.253	44576
Secção sobre o pilar	3.44E-10	6.169	34109
Hipercubo Latino N=100			
Secção a meio vão	5.19E-10	6.104	878154
Secção sobre o pilar	3.2E-10	6.180	1117671

Por último, falta comparar os valores dos índices de fiabilidade obtidos com os valores mínimos recomendados do EC-0 (2002) presentes na tabela 2.2. O valor mais alto que se encontra nessa tabela é 5.2, que está para construções com classe de fiabilidade 3 (RC3) e para o estado limite último com um período de referência de um ano. Ou seja segundo todas as simulações efectuadas, a passagem superior verifica os valores mínimos recomendados, para qualquer situação de dimensionamento estrutural.

4.4. ALTERAÇÃO DAS CARACTERÍSTICAS DA PASSAGEM SUPERIOR

Considera-se que passaram vários anos desde a construção da passagem superior e que não houve qualquer manutenção. Uma equipa de técnicos, é contratada para verificar as condições em que se encontra a estrutura e para determinar a fiabilidade estrutural. Ao terminar as verificações a equipa determinou que houve uma deterioração das armaduras. Na secção a meio vão, estão deteriorados 5 dos 13 varões de aço ordinários existentes na parte inferior. A todo o comprimento, estão deteriorados

2 dos 10 cordões de aço de pré-esforço existente. Esta secção tem como objectivo averiguar se, apesar destas condições, a passagem superior apresenta um índice de fiabilidade admissível. Dada a hiperestaticidade da estrutura, será feita uma análise à segurança estrutural que já incorpora redistribuições de momentos. Para tal serão elaboradas duas simulações, uma pelo método de Monte Carlo e outra pelo método do Hipercubo latino. Para o primeiro método serão simuladas 10000 amostras e para o segundo 100. Na tabela 4.19 estão expostas as variáveis que sofreram alteração.

Tabela 4.19 - Armaduras deterioradas

Armaduras ordinários e de pré-esforço deteriorados					
Aço Ordinário S500					
A_s	- Área da armadura	Normal	*	4.3	*
A_{si}	- Área da armadura inferior da secção a meio vão As nominal = 21.999 cm ²	Normal	20.239 cm ²	4.3	0.87 cm ²
* - As real / As nominal = 0.92					
Aço de Pré-Esforço S1680/1860 – 0.5”					
A_p	- Armadura de pré-esforço (x8)	Normal	0.93 cm ²	2	0.0186 cm ²

Para esta avaliação não basta verificar as secções individualmente, vai ser necessário avaliar a estrutura como um todo, ao contrário da secção 4.3, onde foram efectuadas avaliações de fiabilidade estrutural a secções individualmente, aplicando a combinação de cargas mais gravosa. Então é necessário estudar o comportamento da passagem superior para cada combinação e cada secção.

Sendo assim, nesta avaliação de fiabilidade estrutural, em vez de comparar a resistência de cada secção com o momento actuante na secção, vai-se somar a capacidade resistentes aos momentos negativos da secção sobre o pilar, à capacidade resistente aos momentos positivos da secção a meio vão e verifica-se a fiabilidade estrutural para uma carga geral ($PL^2/8$), onde P é a carga distribuída e L é o comprimento do vão. Então avaliação é dada pela seguinte expressão:

$$M_R^+ + M_R^- \geq M_S \quad (4.1)$$

Onde M_R^+ é o momento resistente positivo a meio vão, M_S^- é o momento resistente na secção sobre o pilar e M_S é o momento actuante provocado pela carga geral. O momento actuante também pode ser calculado, somando os momentos actuantes negativos aos positivos, em módulo, resultando num momento total (expressão 4.2). Os momentos actuantes já se encontram definidos na tabela 4.4.

$$M_S = |M_S^+| + |M_S^-| \quad (4.2)$$

Este método pode-se aplicar, se as cargas forem uniformemente distribuídas e se os momentos actuantes forem iguais nos extremos do vão. A primeira combinação verifica as duas condições, mas a segunda não. Os momentos negativos actuantes da segunda combinação não são iguais, resultando num diagrama inclinado. Para este caso basta encontrar o valor actuante médio.

$$M_S = M_S^+ + \frac{M_S^{-Esquerda} + M_S^{-Direita}}{2} \quad (4.3)$$

A figura 4.16 ilustra o cálculo do momento actuante para a segunda combinação de cargas. Depois de calcular este momentos, a partir das amostras simuladas, procede-se a avaliação da fiabilidade

estrutural. Este processo de avaliação tem uma grande vantagem sobre a avaliação individual das secções, pois incorpora a hiperstaticidade da estrutura. Assim ao efectuar estes cálculos, resulta uma avaliação da fiabilidade, se necessário, com redistribuições de momentos. É importante referir que num caso real, depois se efectuar redistribuições, seria necessário verificar se estas redistribuições não ultrapassam o limite estabelecido. Como o objectivo desta secção é demonstrar a utilidade da ferramenta elaborada e não o cálculo estrutural, essa verificação será ignorada.

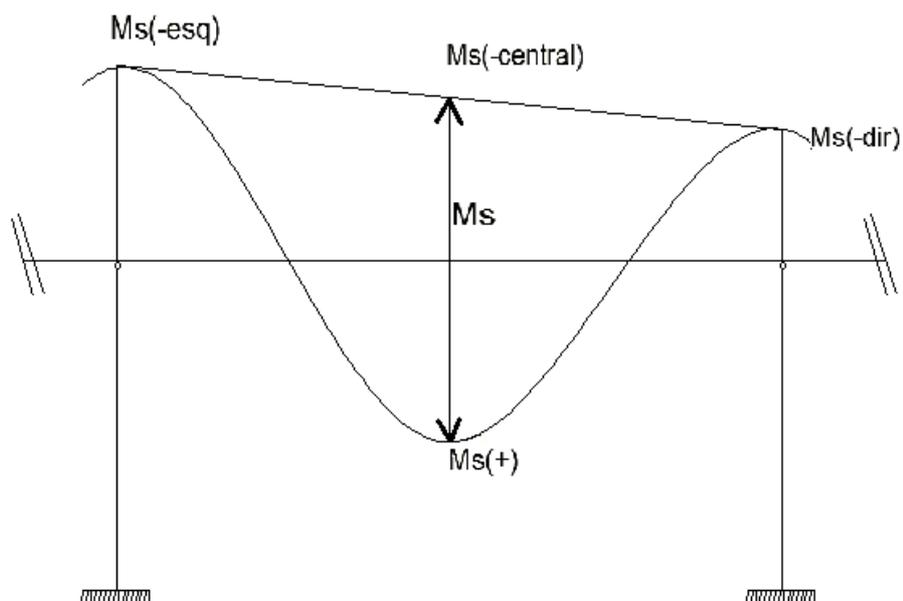


Fig. 4.16 - Representação gráfica do cálculo do momento actuante total

Procede-se à simulação de 10000 e 100 amostras, pelo método de Monte Carlo e Hipercubo Latino, respectivamente.

Na tabela 4.20 estão expostos os momentos resistentes e actuantes nas duas secções para as duas combinações. É de notar que o cálculo do momento actuante da segunda combinação foi calculado segundo a expressão (4.3). Recordando os momentos resistentes com secção intacta, $2668.343kN.m$ (secção a meio vão) e $3420.621kN.m$ (secção sobre o pilar), constata-se que a secção a meio vão, é a mais danificada. Visto que, é a armadura ordinária desta secção que se encontra deteriorada, este resultado era previsto. Tendo em conta que o momento positivo actuante a meio vão é na ordem dos $1621kN.m$, é fácil prever que, para verificar os índices mínimos recomendados pelo EC-0 (2002), vai ser necessário recorrer a redistribuições de momentos.

Comparando os resultados das simulações efectuadas pelos dois métodos aplicados, verifica-se que as médias dos momentos resistentes e actuantes são semelhantes, mas os desvios padrão e valores extremos, já são ligeiramente diferentes. Observando a tabela 4.21, constata-se o mesmo facto da tabela anterior, médias semelhantes e desvios padrão e valores extremos diferentes. Na avaliação de fiabilidade estrutural à passagem superior em boas condições, este facto também se verificou.

Tabela 4.20 - Momentos resistentes e actuantes (Secções deterioradas)

	Média	Mínimo	Máximo	CV(%)	Desvio Padrão
Monte Carlo N=10000					
Mr(+)	1715.32	1457.24	2028.84	4.287	73.538
Mr(-)	3240,32	2626.92	3925.33	5.537	179,42
Mr=Mr(+)+Mr(-)	4955.64	4125.40	5842.70	4.746	235.19
Combinação 1 Ms	3561.83	2901.81	4197.45	4.546	161,91
Combinação 2 Ms	3562.76	3072.80	4120.22	3.792	135.11
Hipercubo Latino N=100					
Mr(+)	1715.66	1528.62	1982.72	4.518	77.51
Mr(-)	3240.25	2856.70	3559.42	5.495	178.06
Mr=Mr(+)+Mr(-)	4955.91	4427.99	5486.76	4.842	239.98
Combinação 1 Ms	3562.60	3089.88	3950,78	4.884	173.99
Combinação 2 Ms	3563.60	3209.60	4029.01	3.951	140.79

Tabela 4.21 - Média, desvio padrão, mínimo e máximo da resposta estrutural (secções deterioradas)

	Z – Média (kN.m)	Z – Mínimo (kN.m)	Z – Máximo (kN.m)	CV(%)	Z - Desvio Padrão (kN.m)
Monte Carlo N=10000					
Combinação 1 Ms	1393.81	394.56	2490.00	20.49	285.53
Combinação 2 Ms	1392.88	451.37	2349.09	19.47	271.23
Hipercubo Latino N=100					
Combinação 1 Ms	1393.31	690.37	2137.88	21.27	296.42
Combinação 2 Ms	1392.31	750.69	1926.47	19.98	278.24

Tabela 4.22 - Resultados finais da avaliação de fiabilidade estrutural (secções deterioradas)

	β	P_F	COV	erro(%)
Monte Carlo N=10000				
Combinação 1 Ms	4.881	527E-07	-	2756.003
Combinação 2 Ms	5,135	1,41E-07	-	5329.671
Hipercubo Latino N=100				
Combinação 1 Ms	4.700	1,30E-06	-	17554.22
Combinação 2 Ms	5,004	2,81E-07	-	37752.85

As medidas de fiabilidade estrutural, índice de fiabilidade e probabilidade de rotura, estão expostas na tabela 4.22. Os índices de fiabilidade obtidos, a partir das duas simulações efectuadas são ligeiramente diferentes, sendo o índice obtido a partir do método do Hipercubo Latino menor. Esta diferença deve-se às diferenças dos desvios padrão. Como os desvios padrão calculados pelo método do Hipercubo Latino, são maiores (neste caso), vão provocar uma distribuição mais achatada. Neste género de distribuição os valores mais extremos têm maior probabilidade de ocorrer, do que em distribuições com desvios padrão pequenos. A comparação entre várias funções de densidade com desvios diferentes está ilustrada na figura 2.3 da secção 2.1.3.2.

Considerando que a passagem superior é uma estrutura da terceira classe de fiabilidade (RC3) e que se pretende verificar se nas condições actuais, a passagem ainda respeita o índice mínimo recomendado pelo EC-0 (2002), exposto na tabela 2.2. Para 50 anos de referência o índice de fiabilidade mínimo recomendado é 4.3. Comparando com os valores obtidos, conclui-se que a avaliação de fiabilidade estrutural à passagem superior deteriorada, cumpre os requisitos mínimos recomendados pelo EC-0 (2002).

5

CONCLUSÃO

5.1. CONCLUSÕES FINAIS

A inovação é um conceito inerente à construção civil. Todos os dias criam-se novas soluções construtivas com intuito de satisfazer a cultura humana. Esta encontra-se em constante transformação, obrigando a construção, a estar um passo a frente. Assim surge uma dúvida. Como tirar proveito de uma solução construtiva nova, para a qual ainda não existem coeficientes de segurança adequados. Pode-se adaptar um coeficiente conhecido, utilizado numa solução semelhante, mas o aproveitamento nunca será ideal. Então é necessário recorrer a metodologias que permitam aplicar a solução construtiva, garantindo segurança estrutural e aproveitamento máximo. Entre muitas metodologias disponíveis, têm-se os métodos de simulação.

Esta dissertação foi desenvolvida com o objectivo principal elaborar um programa de apoio à avaliação de fiabilidade estrutural com base em métodos de simulação, mais concretamente o método de Monte Carlo e a sua variante, método do Hipercubo Latino. Para criar uma base teórica que sustente o programa de cálculo, foram desenvolvidos conceitos de estatística, segurança estrutural e métodos de simulação.

O método de Monte Carlo, também conhecido por método de simulação pura, é caracterizado pela total aleatoriedade. Esta simplicidade de aplicação é uma vantagem, mas também apresenta desvantagens. Este método necessita de demasiadas amostras para obter resultados com precisão admissível, o que pode levar a um esforço computacional excessivo. Outra desvantagem deste método, é a impossibilidade de garantir que as amostras resultantes, correspondam à lei probabilística usada. Devido à total aleatoriedade, ao gerar amostras segundo uma lei probabilística, nada garante que o conjunto de amostras resultante, corresponda a essa mesma lei. Para averiguar se as amostras respeitam essa lei, recorre-se aos seguintes testes de hipótese, teste de normalidade (distribuições normais) e teste de Kolmogorov-Smirnov. Se o conjunto de amostras verificar os testes de hipótese, considera-se um conjunto válido e prossegue-se à avaliação de fiabilidade estrutural. Para reduzir o tempo de computação inerente, desenvolveram-se técnicas de redução de variância, entre as quais se encontra o método de Hipercubo Latino.

Ao contrário do método de Monte Carlo, o método do Hipercubo Latino não tem uma aplicação simples. Este método envolve um algoritmo de simulação mais complexo, que impede a repetição de valores num conjunto de amostras e garante um comportamento segundo a lei probabilística usada. Contudo, apesar de aplicação mais difícil, não necessita de efectuar testes de hipótese. Mas a característica mais importante deste método, que representa uma grande vantagem sobre o método de Monte Carlo, é a elevada precisão, particularmente com amostragens pequenas. Sendo assim, o facto de não necessitar de testes de hipótese e de utilizar menos amostras, fazem com que este método seja mais eficiente.

Com o intuito de criar um programa de apoio flexível e adaptável a vários casos de avaliação fiabilidade, dividiu-se o programa em dois módulos. A simulação de variáveis aleatórias segundo uma lei teórica, dá-se no primeiro módulo. O utilizador tem disponível uma serie de distribuições probabilísticas e os dois métodos de simulação acima referidos. Terminada a simulação, exportam-se os conjuntos de amostras para uma folha Excel. Esta ligação directa com o Excel, facilita a interacção com outros programas de cálculo. No segundo módulo do programa, efectua-se a avaliação à fiabilidade propriamente dita. O utilizador introduz dois vectores, armazenados no Excel, e o programa verifica se uma função de estado limite é respeitada. O primeiro vector caracteriza a função de estado limite e o segundo caracteriza a função actuante. Efectuados os cálculos, o segundo módulo exporta, para Excel, a probabilidade rotura e o índice de fiabilidade. Estas duas medidas de fiabilidade caracterizam a probabilidade da função actuante ser superior à função de estado limite. Deste modo obteve-se um programa de fácil utilização e de aplicação ampla, tal como era pretendido. Qualquer problema que possa ser representado por funções limite pode ser avaliado neste programa de apoio.

Para demonstrar a utilidade do programa elaborado, avaliou-se a fiabilidade estrutural de uma passagem superior, segundo o método de Monte Carlo e o método do Hipercubo Latino. Para o primeiro método simularam-se 10000 e 100000 amostras e para o segundo simularam-se 100. No processo de avaliação de fiabilidade estrutural, constatou-se que o método do Hipercubo Latino consumia, consideravelmente, menos tempo que o primeiro método. Analisando os dados obtidos conclui-se que, com poucas amostras o método do Hipercubo Latino permite obter resultados com precisão admissível, ao contrario do primeiro método. Tendo em conta que a eficiência é um conceito de grande interesse no dimensionamento de estruturas, entre os métodos de simulação estudados, sugere-se o uso do método do Hipercubo Latino para avaliações de fiabilidade estrutural, quando a resposta estrutural tem uma distribuição normal.

Consequentemente, foi possível constatar que o programa elaborado é aplicável à avaliação da fiabilidade estrutural, com capacidade para simular um grande número de variáveis aleatórias segundo um leque variado de distribuições probabilísticas e ainda segundo o método de Monte Carlo ou do Hipercubo Latino.

5.2. DESENVOLVIMENTOS FUTUROS

O programa desenvolvido nesta dissertação foi testado num problema estrutural simples, recorrendo a dois métodos de avaliação de fiabilidade e a um tipo de distribuição. Os seguintes aspectos podem vir a ser desenvolvidos:

- Aplicação do programa de apoio a um caso mais realista que inclua análise de estruturas mais avançadas, de comportamento não linear.
- Proceder à verificação de uma função de estado limite, baseada no comportamento dinâmico das estruturas.
- Aplicar outros tipos de distribuições a simulações de variáveis aleatórias, dado que só se aplicou a distribuição normal.
- Complementar o programa de apoio à avaliação de fiabilidade estrutural com um programa de cálculo automático generalizado, capaz de importar amostras que caracterizam os materiais e a geometria, e que exporte um conjunto de amostras representativo da função de estado limite.

Nesta dissertação foram estudados dois métodos de avaliação de fiabilidade estrutural, contudo existem mais métodos que podem ser estudados e aplicados a ferramentas informáticas. A elaboração

de um programa de apoio à avaliação da fiabilidade estrutural, baseada em outros métodos, permitiria um estudo comparativo mais preciso. Este estudo poderia caracterizar qual o tipo de metodologia de avaliação seria mais adequado a sistemas estruturais, caracterizando a sua complexidade e eficiência.

6

BIBLIOGRAFIA

- Ayyub, B.M., Guran, A., Haldar, A., Zadeh, L. (1997). *Uncertainty modeling in vibration, control and fuzzy analysis of structural systems*, World Scientific, Singapore.
- Binder, K., (1979). *Monte Carlo Methods in Statistical Physic*. Springer-Verlag, Berlin.
- Delgado, J.M. (2002). *Avaliação de segurança de estruturas reticuladas com comportamento não linear material e geométrico*. Dissertação para Doutoramento em Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Porto, disponível em <http://repositorio-aberto.up.pt/handle/10216/10957>.
- Comité Europeu de Normalização, (2002). *EN1990: Eurocode - Basis of Structural Design*. CEN, Brussels, final version.
- Comité Europeu de Normalização, (2002). *EN1991-1-1: Eurocode 1: Action on Structures - Part 1-1: General Actions - Densities, Self-weight, Imposed Loads for Buildings*. CEN, Brussels, final version.
- Cornell, A. C. (1969). *A Probability-Based Structural Code*. Journal Amer. Concrete Inst., Vol. 66, n.º 12, pp.974-985
- Guimarães, R.C., Cabral, J.A.S. (1997). *Estatística – Edição Revista*. McGraw-Hill, Faculdade de Engenharia da Universidade do Porto.
- Henriques, A.A.R. (1998). *Aplicação de Novos Conceitos de Segurança no Dimensionamento do Betão Estrutural*. Dissertação para Doutoramento em Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Porto, disponível em <http://paginas.fe.up.pt/~aarh/tese-A4.pdf>.
- Henriques, A.A.R., Calheiros, F., Figueiras, J. (1995) *Modelo de cálculo da fiabilidade estrutural*, IV Encontro Nacional de Mecânica Computacional, 10 de Abril de 1995, Lisboa, pp.15-26.
- Haldar, A. & Mahadevan, S. (2000). *Probability, Reliability and Statistical Methods in Engineering Design*. John Wiley & Sons, New York.
- Haldar, A., (2006). *Recent Developments in Reliability-Based Civil Engineering*. World Scientific, Singapore.
- Blanchard, J., Canal no Youtube (<http://www.youtube.com/user/jakeblanchard>). 25/03/2010.
- Laranja, R., Estevão, J. (2000) *Simulação Numérica na Avaliação da Fiabilidade de Estruturas de Betão Armado*. Encontro Nacional Betão Estrutural, 2000, Faculdade de Engenharia da Universidade do Porto, pp. 295-301, Porto, disponível em <http://hdl.handle.net/10400.1/91>.
- Laranja, R. e Brito, J. (2003). *Verificação Probabilística da Segurança das Estruturas*. Revista Engenharia Civil – UM, n.º 18, pp. 63-79, disponível em <http://hdl.handle.net/10400.1/48>.

Mathworks, the (http://www.mathworks.com/access/helpdesk/help/toolbox/stats/bq_w_hm.html#bq_w_ie-7). 01/04/2010 a 27/06/2010.

Matlab, Canal no Youtube (<http://www.youtube.com/user/MATLAB>). 25/03/2010.

Melchers, R.E. (1999). *Structural Reliability Analysis and Prediction*. 2ª edição. John Wiley & Sons, Chichester.

Nowak, A.S. & Collins, K.R. (2000). *Reliability of Structures*. McGraw-Hill, New York.

Ministério da Habitação, Obras Públicas e Transportes (1983). *Regulamento de Segurança e Acções para Estruturas de Edifícios e Pontes*. Ministério da Habitação, Obras Públicas e Transportes, Lisboa.

Thoft-Christensen, P., Murotsu, Y. (1986). *Application of structural systems reliability theory*. Springer-Verlag, Berlin.

Veiga, José M.G.C. (2008). *Métodos da Análise das Incertezas na Verificação da Segurança Estrutural em Engenharia Civil*. Dissertação para Doutoramento em Ciências de Engenharia, Faculdade de Engenharia da Universidade do Porto, Porto, disponível em <http://repositorio-aberto.up.pt/bitstream/10216/11644/4/Texto%20integral.pdf>.

Wiśniewski, D.F. (2007). *Safety Formats for the Assessment of Concrete Bridges - with special focus on precast concrete*. Dissertação para doutoramento em Engenharia Civil, Universidade do Minho, Guimarães, disponível em <http://hdl.handle.net/1822/6753>.

ANEXOS

ANEXO 1 - CÓDIGO DO PROGRAMA DE APOIO

No presente anexo está exposto o código do programa de apoio à avaliação de fiabilidade estrutural. Este código foi retirado directamente do ficheiro gerado pelo “software” Matlab. Serão adicionados alguns comentários que ajudaram a compreender o funcionamento do programa elaborado. Todos os comentários introduzidos estarão **sombreados** e antecidos por um símbolo de percentagem “%”. Deste modo é possível copiar este texto para um ficheiro Matlab e correr o programa de apoio.

Código do programa (m-file):

```
function varargout = v01(varargin)

% Definições Básicas do programa-----
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @v01_OpeningFcn, ...
                  'gui_OutputFcn',  @v01_OutputFcn, ...
                  'gui_LayoutFcn',   [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function v01_OpeningFcn(hObject, eventdata, handles, varargin)

% Início do Programa----- Declaração de variáveis globais-----
handles.seed=sum(100*clock);
handles.eseed=0;
RandStream.setDefaultStream(RandStream('mt19937ar','seed',handles.seed));
handles.matrix=0;
handles.matrix_iterator=1;
handles.flag1= 0;
handles.flag2= 0;
handles.matrix_out=0;
handles.vnomes={' '};
handles.mresistente=0;
handles.msolicitador=0;
handles.msistema=0;
handles.vnomesfinal=' ' ;
handles.vresultados=0;
handles.uz=0;
handles.vz=0;
handles.vhipotese=' ' ;
handles.vhipoteseks=0;
handles.vkserror=0;
```

```
handles.vhipotesemod2=['e' 'e'];
handles.vhipoteseksmod2=[0 0];
guidata(hObject, handles);
% Declarações dos espaços editáveis do primeiro módulo-----

function varargout = v01_OutputFcn(hObject, eventdata, handles)

function pop_Callback(hObject, eventdata, handles)

function pop_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function nsamples_Callback(hObject, eventdata, handles)

function nsamples_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edt1_Callback(hObject, eventdata, handles)

function edt1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edt2_Callback(hObject, eventdata, handles)

function edt2_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edt3_Callback(hObject, eventdata, handles)

function edt3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu_Callback(hObject, eventdata, handles)

% Lista dos "Tipos de Distribuição"-----
----Definição dos textos que aparecem por baixo dos espaços editáveis ----

str = get(hObject, 'String');
val = get(hObject, 'Value');
switch str{val};
    case 'Normal'
        set(handles.txt1, 'String', 'Média');
        set(handles.txt2, 'String', 'Desvio Padrão');
```

```

set(handles.edt2,'Visible','On');
set(handles.txt3,'String','');
set(handles.edt3,'Visible','Off');
set(handles.txt11,'String','');
set(handles.txt22,'String','');
set(handles.txt33,'String','');

case 'LogNormal'
set(handles.txt1,'String','Mu');
set(handles.txt2,'String','Sigma');
set(handles.edt2,'Visible','On');
set(handles.txt3,'String','');
set(handles.edt3,'Visible','Off');
set(handles.txt11,'String','Média da distribuição normal
associada');
set(handles.txt22,'String','Desvio padrão da distribuição normal
associada');
set(handles.txt33,'String','');
case 'Beta'
set(handles.txt1,'String','A');
set(handles.txt2,'String','B');
set(handles.edt2,'Visible','On');
set(handles.txt3,'String','');
set(handles.edt3,'Visible','Off');
set(handles.txt11,'String','A > 0');
set(handles.txt22,'String','B > 0');
set(handles.txt33,'String','');
case 'Extremos I'
set(handles.txt1,'String','Mu');
set(handles.txt2,'String','Sigma');
set(handles.edt2,'Visible','On');
set(handles.txt3,'String','');
set(handles.edt3,'Visible','Off');
set(handles.txt11,'String','Parâmetro de localização');
set(handles.txt22,'String','Parâmetro escalar');
set(handles.txt33,'String','');
case 'Extremos II'
set(handles.txt1,'String','Mu');
set(handles.txt2,'String','Sigma');
set(handles.edt2,'Visible','On');
set(handles.txt3,'String','K');
set(handles.edt3,'Visible','On');
set(handles.txt11,'String','Parâmetro de localização');
set(handles.txt22,'String','Parâmetro escalar');
set(handles.txt33,'String','Parâmetro de forma, K > 0');
case 'Extremos III'
set(handles.txt1,'String','Mu');
set(handles.txt2,'String','Sigma');
set(handles.edt2,'Visible','On');
set(handles.txt3,'String','K');
set(handles.edt3,'Visible','On');
set(handles.txt11,'String','Parâmetro de localização');
set(handles.txt22,'String','Parâmetro escalar');
set(handles.txt33,'String','Parâmetro de forma, K < 0');
case 'Binomial'
set(handles.txt1,'String','N');
set(handles.txt2,'String','P');
set(handles.edt2,'Visible','On');
set(handles.txt3,'String','');

```

```
        set(handles.edt3,'Visible','Off');
        set(handles.txt11,'String','N tentativas');
        set(handles.txt22,'String','Probabilidade de sucesso [0;1]');
        set(handles.txt33,'String','');
    case 'Geometrica'
        set(handles.txt1,'String','P');
        set(handles.txt2,'String','');
        set(handles.edt2,'Visible','Off');
        set(handles.txt3,'String','');
        set(handles.edt3,'Visible','Off');
        set(handles.txt11,'String','Probabilidade de sucesso [0;1]');
        set(handles.txt22,'String','');
        set(handles.txt33,'String','');
    case 'Binomial Negativa'
        set(handles.txt1,'String','k');
        set(handles.txt2,'String','P');
        set(handles.edt2,'Visible','On');
        set(handles.txt3,'String','');
        set(handles.edt3,'Visible','Off');
        set(handles.txt11,'String','k > 0');
        set(handles.txt22,'String','Probabilidade de sucesso [0;1]');
        set(handles.txt33,'String','');
    case 'Poisson'
        set(handles.txt1,'String','Lambda');
        set(handles.txt2,'String','');
        set(handles.edt2,'Visible','Off');
        set(handles.txt3,'String','');
        set(handles.edt3,'Visible','Off');
        set(handles.txt11,'String','Taxa média de ocorrências');
        set(handles.txt22,'String','');
        set(handles.txt33,'String','');

end

function popupmenu_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function lbl_Callback(hObject, eventdata, handles)

function lbl_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edtnome_Callback(hObject, eventdata, handles)

function edtnome_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pbseed_Callback(hObject, eventdata, handles)
```

```

%botão seed----- Início-----
-----sobrepor a semente-----

handles.seed=str2double(get(handles.edtseed,'String'));
fprintf('seed= %d\n', handles.seed);
handles.eseed=handles.seed;
guidata(hObject, handles);
RandStream.setDefaultStream(RandStream('mt19937ar','seed',handles.seed));
set(handles.edtseed,'Enable','Off');
set(handles.pbseed,'Enable','Off');
guidata(hObject, handles);

function pb1_Callback(hObject, eventdata, handles)
% Botão "Adicionar"-----Início-----
--Verificação inicial dos dados introduzidos-----

edt11=str2double(get(handles.nsamples,'String'));
edt11=edt11-edt11;
if edt11==0
else
    errordlg('Dados incompatíveis!!! Altere os dados introduzidos no espaço
"N".')
    dbstop
end
edt11=str2double(get(handles.edt1,'String'));
edt11=edt11-edt11;
if edt11==0
else
    errordlg('Dados incompatíveis!!! Verifique os Parâmetros de
distribuição.')
```

```

    dbstop
end
edt11=str2double(get(handles.edt2,'String'));
edt11=edt11-edt11;
if edt11==0
else
    errordlg('Dados incompatíveis!!! Verifique os Parâmetros de
distribuição.')
```

```

    dbstop
end
% Se houver algum erro nos dados introduzidos, interrompe o programa e pede
reintrodução de dados-----
--Procede com as acções necessárias para efectuar as simulações-----

set(handles.pb1,'Enable','Off');
set(handles.pbgerar,'Enable','Off');
set(handles.popupmenu,'Enable','Off');
set(handles.pop2,'Enable','Off');
str = get(handles.popupmenu, 'String');
val = get(handles.popupmenu, 'Value');
str1 = get(handles.pop2, 'String');
vall = get(handles.pop2, 'Value');
limite2=str2double(get(handles.nsamples,'String'));
x=handles.matrix_iterator
if x==1
    handles.vnomes=get(handles.edtnome,{'String'});
else
    handles.vnomes(x)=get(handles.edtnome,{'String'});
end

```

```

end
handles.vksexerror=0;
ksflag=0;
switch str1{val1};
    case 'Monte Carlo'
% Simulação pelo método de Monte Carlo-----

        contadorvezes=0;
        while handles.vksexerror<2
            contadorvezes=contadorvezes+1;
            if contadorvezes>10
% Se a simulação for repetida mais de 10 vezes o programa pede ao
utilizador para alterar os parâmetros de distribuição introduzidos-----

                errordlg('Dados incompatíveis!!! Verifique os Parâmetros de
distribuição ou experimente aumentar o número de amostras.')
                set(handles.pb1,'Enable','On');
                set(handles.pbgerar,'Enable','On');
                set(handles.popupmenu,'Enable','On');
                set(handles.pop2,'Enable','On');
                dbstop
            else
                switch str{val};
                case 'Normal'
% Simulação segundo a lei normal-----

                    flagnorm=1;
                    iterator = 1;
                    V=0;
                    while(iterator<=limite2)
                        V(iterator)=norminv(rand,str2double(get(handles.edt1,'String')),str2d
double(get(handles.edt2,'String')));
                        iterator=iterator+1;
                    end
                    case 'LogNormal'
% Simulação segundo a lei lognormal-----

                        flagnorm=1;
                        iterator = 1;
                        V=0;
                        while(iterator<=limite2)
                            V(iterator)=logninv(rand,str2double(get(handles.edt1,'String')),str2d
ouble(get(handles.edt2,'String')));
                            iterator=iterator+1;
                        end
                    case 'Beta'
% Simulação segundo a lei Beta-----
---Antes verifica se existem erros, específicos a esta lei, nos dados
introduzidos-----

                        edta=str2double(get(handles.edt1,'String'));
                        edtb=str2double(get(handles.edt2,'String'));
                        if edta>0
                            else
                                errordlg('Dados incompatíveis!!! A > 0!!!!')
                                set(handles.pb1,'Enable','On');
                                set(handles.pbgerar,'Enable','On');
                                set(handles.popupmenu,'Enable','On');

```

```

        set(handles.pop2, 'Enable', 'On');
        dbstop
    end
    if edtb>0
    else
        errordlg('Dados incompatíveis!!! B > 0!!!')
        set(handles.pb1, 'Enable', 'On');
        set(handles.pbgerar, 'Enable', 'On');
        set(handles.popupmenu, 'Enable', 'On');
        set(handles.pop2, 'Enable', 'On');
        dbstop
    end
    flagnorm=1;
    iterator = 1;
    V=0;
    while(iterator<=limite2)
        V(iterator)=betainv(rand, str2double(get(handles.edt1, 'String')), str2double(get(handles.edt2, 'String')));
        iterator=iterator+1;
    end
case 'Extremos I'
% Simulação segundo a lei de Extremos I-----

        flagnorm=1;
        iterator = 1;
        V=0;
        while(iterator<=limite2)
            V(iterator)=gevinv(rand, 0, str2double(get(handles.edt2, 'String')), str2double(get(handles.edt1, 'String')));
            iterator=iterator+1;
        end
case 'Extremos II'
% Simulação segundo a lei de Extremos II-----
-Antes verifica se existem erros nos dados introduzidos, específicos a esta lei, -----

        edt11=str2double(get(handles.edt3, 'String'));
        edt11=edt11-edt11;
        if edt11==0
        else
            errordlg('Dados incompatíveis!!! Verifique os Parâmetros de distribuição.')
            set(handles.pb1, 'Enable', 'On');
            set(handles.pbgerar, 'Enable', 'On');
            set(handles.popupmenu, 'Enable', 'On');
            set(handles.pop2, 'Enable', 'On');
            dbstop
        end
        edtk=str2double(get(handles.edt3, 'String'));
        if edtk>0
        else
            errordlg('Dados incompatíveis!!! K > 0!!!')
            set(handles.pb1, 'Enable', 'On');
            set(handles.pbgerar, 'Enable', 'On');
            set(handles.popupmenu, 'Enable', 'On');
            set(handles.pop2, 'Enable', 'On');
            dbstop
        end
end

```

```

flagnorm=1;
iterator = 1;
V=0;
while(iterator<=limite2)
    V(iterator)=gevinv(rand, str2double(get(handles.edt3, 'String')), str2double(get(handles.edt2, 'String')), str2double(get(handles.edt1, 'String')));
    iterator=iterator+1;
end
case 'Extremos III'
% Simulação segundo a lei de Extremos III-----
---Antes verifica se existem erros nos dados introduzidos, específicos a esta lei, -----

    edt11=str2double(get(handles.edt3, 'String'));
    edt11=edt11-edt11;
    if edt11==0
    else
        errordlg('Dados incompatíveis!!! Verifique os Parâmetros de distribuição.')
        set(handles.pb1, 'Enable', 'On');
        set(handles.pbgerar, 'Enable', 'On');
        set(handles.popupmenu, 'Enable', 'On');
        set(handles.pop2, 'Enable', 'On');
        dbstop
    end
    edtk=str2double(get(handles.edt3, 'String'));
    if edtk<0
    else
        errordlg('Dados incompatíveis!!! K < 0!!!')
        set(handles.pb1, 'Enable', 'On');
        set(handles.pbgerar, 'Enable', 'On');
        set(handles.popupmenu, 'Enable', 'On');
        set(handles.pop2, 'Enable', 'On');
        dbstop
    end
    flagnorm=1;
    iterator = 1;
    V=0;
    while(iterator<=limite2)
        V(iterator)=gevinv(rand, str2double(get(handles.edt3, 'String')), str2double(get(handles.edt2, 'String')), str2double(get(handles.edt1, 'String')));
        iterator=iterator+1;
    end
case 'Binomial'
% Simulação segundo a lei Binomial-----

    handles.vkerror=3;
    flagnorm=2;
    iterator = 1;
    V=0;
while(iterator<=limite2)
    V(iterator)=binoinv(rand, str2double(get(handles.edt1, 'String')), str2double(get(handles.edt2, 'String')));
    iterator=iterator+1;
end
case 'Geometrica'
% Simulação segundo a lei Geometrica-----

```

```

        handles.vkerror=3;
        flagnorm=2;
        iterator = 1;
        V=0;
        while(iterator<=limite2)
            V(iterator)=geoinv(rand, str2double(get(handles.edt1, 'String')));
            iterator=iterator+1;
        end
    case 'Binomial Negativa'
% Simulação segundo a lei Binomial negativa-----

        handles.vkerror=3;
        flagnorm=2;
        iterator = 1;
        V=0;
        while(iterator<=limite2)
            V(iterator)=nbininv(rand, str2double(get(handles.edt1, 'String')), str2d
            ouble(get(handles.edt2, 'String')));
            iterator=iterator+1;
        end
    case 'Poisson'
% Simulação segundo a lei de Poisson-----

        handles.vkerror=3;
        flagnorm=2;
        iterator = 1;
        V=0;
        while(iterator<=limite2)
            V(iterator)=poissinv(rand, str2double(get(handles.edt1, 'String')));
            iterator=iterator+1;
        end
    end
end
%-----Teste de Normalidade-----Distribuição Normal-----
switch str{val};
    case 'Normal'
        if handles.vkerror==0
            flagcond=0;
            hipotese='';
            norm11=0;
            norm12=0;
            norm21=0;
            norm22=0;
            norm1=0;
            norm2=0;
            condi1=0;
            condi2=0;
            condi22=0;
            condi3=0;

            condi44=0;
            desvio1=0;
            desvio2=0;
            for i=1:1:limite2
norm11=norm11+(V(i)-str2double(get(handles.edt1, 'String')))^3;
norm12=norm12+(V(i)-str2double(get(handles.edt1, 'String')))^2 ;

norm21=norm21+(V(i)-str2double(get(handles.edt1, 'String')))^4;

```

```

norm22=norm22+(V(i)-str2double(get(handles.edt1,'String')))^2 ;
end
norm1=(1/limite2*norm11)/(1/limite2*norm12)^(3/2);
fprintf('norm1= %d\n', norm1);
desvio1=sqrt((6*(limite2-2))/((limite2+1)*(limite2+3)));
fprintf('desvio1= %d\n', desvio1);
norm2=((1/limite2*norm21)/(1/limite2*norm22)^(2))-3;
fprintf('norm2= %d\n', norm2);
desvio2=sqrt((24*limite2*(limite2-2)*(limite2-3))/((limite2+1)^2*(limite2+3)*(limite2+5)));
fprintf('desvio2= %d\n', desvio2);
if (abs(norm1)<1.5*desvio1 && (abs(norm2+6/(limite2+1))<1.5*desvio2))
    hipotese='a';
elseif( (abs(norm1)<2*desvio1 && (abs(norm2+6/(limite2+1))<2*desvio2))
    hipotese='d';
else
    hipotese='r';
end
guidata(hObject, handles);
flaghipotese=handles.matrix_iterator
if flaghipotese==1
    handles.vhipotese=hipotese;
else
    handles.vhipotese(flaghipotese)=hipotese;
end
guidata(hObject, handles);
fprintf('hipotese= %s\n', hipotese);
fprintf('handles.vhipotese= %s\n', handles.vhipotese);
end
%FIM DO TESTE DE NORMALIDADE-----Distribuição Normal-----
%TESTE DE KOLMOGOROV-SMIRNOV-----Distribuição Normal-----
switch hipotese
    case 'a'
        msg=['A Variável "',get(handles.edtnome,'String'),' " verifica as duas
condições do teste de normalidade. Aceita-se a hipótese de normalidade!'];
        msgbox(msg,'Rangen 2.0');
        handles.vkserror=3;
        flaghipoteseks=handles.matrix_iterator
        if flaghipoteseks==1
            handles.vhipoteseks=-1;
        else
            handles.vhipoteseks(flaghipoteseks)=-1;
        end
        guidata(hObject, handles);
    case 'r'
        ksflag=1
        handles.vkserror=0;
    case 'd'
        Vks=sort(V);
        Vteorico=Vks;
        for ks=1:limite2
            vksf(ks)=ks/(limite2+1);
        end
        yteorico=normcdf(Vteorico,str2double(get(handles.edt1,'String')),str2
double(get(handles.edt2,'String')));

        Dn=max(abs(yteorico-vksf));
        Dnn=Dn*(sqrt(str2double(get(handles.nsamples,'String'))))

```

```

fprintf('Dn= %d\n', Dn);
fprintf('Dnn= %d\n', Dnn);
if Dnn>1.36428
    ksflag=2;
    handles.vkerror=0;
else
    handles.vkerror=3;
    msg=['A Variável "',get(handles.edtnome,'String'),' " ficou em
dúvida no teste de normalidade, mas verificou a condição do teste de
Kolmogorov-Smirnov. Aceita-se a simulação!'];
    msgbox(msg,'Rangen 2.0');
    end
flaghipoteseks=handles.matrix_iterator
if flaghipoteseks==1
    handles.vhipoteseks=Dnn;
else
    handles.vhipoteseks(flaghipoteseks)=Dnn;
end
guidata(hObject, handles);
end

%----- Teste de KOLMOGOROV-SMIRNOV para as restantes distribuições-----
otherwise
if flagnorm==1
    ksflag=1
    handles.vkerror=1;
    Vks=sort(V);
    Vteorico=Vks;
for ks=1:limite2
    vkssf(ks)=ks/(limite2+1);
end
switch str{val}
case 'LogNormal'
    yteorico=logncdf(Vteorico,str2double(get(handles.edt1,'String')
),str2double(get(handles.edt2,'String')));
case 'Beta'
    yteorico=betacdf(Vteorico,str2double(get(handles.edt1,'String')
),str2double(get(handles.edt2,'String')));
case 'Extremos I'
    yteorico=gevcdf(Vteorico,0,str2double(get(handles.edt2,'String')
),str2double(get(handles.edt1,'String')));
case 'Extremos II'
    yteorico=gevcdf(Vteorico,str2double(get(handles.edt3,'String')
),str2double(get(handles.edt2,'String')),str2double(get(handles.
edt1,'String')));
case 'Extremos III'
    yteorico=gevcdf(Vteorico,str2double(get(handles.edt3,'String')
),str2double(get(handles.edt2,'String')),str2double(get(handles.
edt1,'String')));
end
Dn=max(abs(yteorico-vkssf));
Dnn=Dn*(sqrt(str2double(get(handles.nsamples,'String'))))
fprintf('Dn= %d\n', Dn);
fprintf('Dnn= %d\n', Dnn);
if Dnn>1.36428
    ksflag=2;
    handles.vkerror=1;
else
handles.vkerror=3;

```

```

        msg=['A Variável "',get(handles.edtnome,'String'),' " verifica o teste
        de Kolmogorov-Smirnov. Aceita-se a simulação!'];
        msgbox(msg,'Rangen 2.0');
    end
    flaghipotese=handles.matrix_iterator
    if flaghipotese==1
        handles.vhipotese='n';
    else
        handles.vhipotese(flaghipotese)='n';
    end
    flaghipoteseks=handles.matrix_iterator
    if flaghipoteseks==1
        handles.vhipoteseks=Dnn;
    else
        handles.vhipoteseks(flaghipoteseks)=Dnn;
    end
    guidata(hObject, handles);
    else
        flaghipotese=handles.matrix_iterator
        if flaghipotese==1
            handles.vhipotese='n';
        else
            handles.vhipotese(flaghipotese)='n';
        end
        flaghipoteseks=handles.matrix_iterator
        if flaghipoteseks==1
            handles.vhipoteseks=-1;
        else
            handles.vhipoteseks(flaghipoteseks)=-1;
        end
    end
end
end
end
%-----FIM DO TESTE DE KOLMOGOROV-SMIRNOV-----
end
end
%-----Simulação pelo método de Hipercubo Latino-----

case 'Hipercubo Latino'
    switch str{val};
        case 'Normal'
% Simulação segundo a lei normal-----

            iterator = 1;
            step=1000000/(str2double(get(handles.nsamples,'String')));
            interval=step/2;
            V=0;
            Bic=0;
            while (interval < 1000000)
                Bic(iterator)=interval/10000;
                V(iterator)=norminv(interval/1000000,str2double(get(handles.edt
                    1,'String')),str2double(get(handles.edt2,'String')));
                interval=interval+step;
                iterator=iterator+1;
            end;
            case 'LogNormal'
% Simulação segundo a lei lognormal-----

                iterator = 1;

```

```

step=1000000/(str2double(get(handles.nsamples,'String')));
interval=step/2;
V=0;
Bic=0;
while (interval < 1000000)
Bic(iterator)=interval/10000;
    V(iterator)=logninv(interval/1000000,str2double(get(handles.edt
    1,'String')),str2double(get(handles.edt2,'String')));
interval=interval+step;
iterator=iterator+1;
end;
case 'Beta'
% Simulação segundo a lei Beta-----
---Antes verifica se existem erros nos dados introduzidos, específicos a
esta lei -----
    edta=str2double(get(handles.edt1,'String'));
    edtb=str2double(get(handles.edt2,'String'));
    if edta>0
    else
        errordlg('Dados incompatíveis!!! A > 0!!!')
        set(handles.pb1,'Enable','On');
        set(handles.pbgerar,'Enable','On');
        set(handles.popupmenu,'Enable','On');
        set(handles.pop2,'Enable','On');
        dbstop
    end
    if edtb>0
    else
        errordlg('Dados incompatíveis!!! B > 0!!!')
        set(handles.pb1,'Enable','On');
        set(handles.pbgerar,'Enable','On');
        set(handles.popupmenu,'Enable','On');
        set(handles.pop2,'Enable','On');
        dbstop
    end
    iterator = 1;
    step=1000000/(str2double(get(handles.nsamples,'String')));
    interval=step/2;
    V=0;
    Bic=0;
    while (interval < 1000000)
    Bic(iterator)=interval/10000;
        V(iterator)=betainv(interval/1000000,str2double(get(handles.edt
        1,'String')),str2double(get(handles.edt2,'String')));
    interval=interval+step;
    iterator=iterator+1;
    end;
case 'Extremos I'
    iterator = 1;
    step=1000000/(str2double(get(handles.nsamples,'String')));
    interval=step/2;
    V=0;
    Bic=0;
    while (interval < 1000000)
    Bic(iterator)=interval/10000;
        V(iterator)=gevinv(interval/1000000,0,str2double(get(handles.ed
        t2,'String')),str2double(get(handles.edt1,'String')));
    interval=interval+step;
    iterator=iterator+1;

```

```

end;
case 'Extremos II'
% Simulação segundo a lei de Extremos II-----
-Antes verifica se existem erros nos dados introduzidos, específicos a esta
lei, -----

    edt11=str2double(get(handles.edt3,'String'));
    edt11=edt11-edt11;
    if edt11==0
    else
        errordlg('Dados incompatíveis!!! Verifique os Parâmetros
de distribuição.')
        set(handles.pb1,'Enable','On');
        set(handles.pbgerar,'Enable','On');
        set(handles.popupmenu,'Enable','On');
        set(handles.pop2,'Enable','On');
        dbstop
    end
    edtk=str2double(get(handles.edt3,'String'));
    if edtk>0
    else
        errordlg('Dados incompatíveis!!! K > 0!!!')
        set(handles.pb1,'Enable','On');
        set(handles.pbgerar,'Enable','On');
        set(handles.popupmenu,'Enable','On');
        set(handles.pop2,'Enable','On');
        dbstop
    end
    iterator = 1;
    step=1000000/(str2double(get(handles.nsamples,'String')));
    interval=step/2;
    V=0;
    Bic=0;
    while (interval < 1000000)
    Bic(iterator)=interval/10000;
        V(iterator)=gevinv(interval/1000000,str2double(get(handles.edt3
,'String')),str2double(get(handles.edt2,'String')),str2double(g
et(handles.edt1,'String')));
        interval=interval+step;
        iterator=iterator+1;
    end;
case 'Extremos III'
% Simulação segundo a lei de Extremos II-----
-Antes verifica se existem erros nos dados introduzidos, específicos a esta
lei, -----

    edt11=str2double(get(handles.edt3,'String'));
    edt11=edt11-edt11;
    if edt11==0
    else
        errordlg('Dados incompatíveis!!! Verifique os Parâmetros de
distribuição.')
        set(handles.pb1,'Enable','On');
        set(handles.pbgerar,'Enable','On');
        set(handles.popupmenu,'Enable','On');
        set(handles.pop2,'Enable','On');
        dbstop
    end
    edtk=str2double(get(handles.edt3,'String'));
    if edtk<0

```

```

else
    errordlg('Dados incompatíveis!!! K < 0!!!')
    set(handles.pb1, 'Enable', 'On');
    set(handles.pbgerar, 'Enable', 'On');
    set(handles.popupmenu, 'Enable', 'On');
    set(handles.pop2, 'Enable', 'On');
    dbstop
end
iterator = 1;
step=1000000/(str2double(get(handles.nsamples, 'String')));
interval=step/2;
V=0;
Bic=0;
while (interval < 1000000)
Bic(iterator)=interval/10000;
    V(iterator)=gevinv(interval/1000000, str2double(get(handles.edt3
    , 'String')), str2double(get(handles.edt2, 'String')), str2double(g
    et(handles.edt1, 'String')));
interval=interval+step;
iterator=iterator+1;
end;
case 'Binomial'
% Simulação segundo a lei Binomial-----

iterator = 1;
step=1000000/(str2double(get(handles.nsamples, 'String')));
interval=step/2;
V=0;
Bic=0;
while (interval < 1000000)
Bic(iterator)=interval/10000;
    V(iterator)=binoinv(interval/1000000, str2double(get(handles.edt
    1, 'String')), str2double(get(handles.edt2, 'String')));
interval=interval+step;
iterator=iterator+1;
end;
case 'Geométrica'
% Simulação segundo a lei Geométrica-----

iterator = 1;
step=1000000/(str2double(get(handles.nsamples, 'String')));
interval=step/2;
V=0;
Bic=0;
while (interval < 1000000)
Bic(iterator)=interval/10000;
    V(iterator)=geoinv(interval/1000000, str2double(get(handles.edt1
    , 'String')));
interval=interval+step;
iterator=iterator+1;
end;
case 'Binomial Negativa'
% Simulação segundo a lei Binomial Negativa-----

iterator = 1;
step=1000000/(str2double(get(handles.nsamples, 'String')));
interval=step/2;
V=0;

```

```

Bic=0;
while (interval < 1000000)
Bic(iterator)=interval/10000;
    V(iterator)=nbininv(interval/1000000, str2double(get(handles.edt
    1, 'String')), str2double(get(handles.edt2, 'String')));
    interval=interval+step;
    iterator=iterator+1;
end;
case 'Poisson'
% Simulação segundo a lei de Poisson-----

    iterator = 1;
    step=1000000/(str2double(get(handles.nsamples, 'String')));
    interval=step/2;
    V=0;
    Bic=0;
    while (interval < 1000000)
    Bic(iterator)=interval/10000;
        V(iterator)=poissinv(interval/1000000, str2double(get(handles.ed
        t1, 'String')));
        interval=interval+step;
        iterator=iterator+1;
    end;
end
flaghipotese=handles.matrix_iterator;
    if flaghipotese==1
        handles.vhipotese='n';
    else
        handles.vhipotese(flaghipotese)='n';
    end
flaghipoteseks=handles.matrix_iterator
    if flaghipoteseks==1
        handles.vhipoteseks=-1;
    else
        handles.vhipoteseks(flaghipoteseks)=-1;
    end
    guidata(hObject, handles);
end
% Acções finais da simulação-----
--Armazenamento dos amostras simuladas numa matriz de saída-----

if (handles.flag1==0)
    handles.matrix=V;
    handles.flag1=1;
    set(handles.nsamples, 'Enable', 'off');
    handles.matrix_iterator=handles.matrix_iterator+1;
else
    handles.matrix=[handles.matrix ; V];
    handles.matrix_iterator=handles.matrix_iterator+1;
end

set(handles.txtnr, 'String', handles.matrix_iterator-1);
guidata(hObject, handles);

    set(handles.pb1, 'Enable', 'On');
    set(handles.pbgerar, 'Enable', 'On');
    set(handles.popupmenu, 'Enable', 'On');

```

```

function pbgerar_Callback(hObject, eventdata, handles)
% Botão "Gerar"-----Início-----
--Exportar Matriz com amostras simuladas para Excel-----

limite1=handles.matrix_iterator-1;
guidata(hObject, handles);
limite2=str2double(get(handles.nsamples,'String'));
string = get(handles.pop2, 'String');
value = get(handles.pop2,'Value');
n=0;
switch string{value};
    case 'Monte Carlo'
        handles.matrix=handles.matrix';
        handles.matrix_out=handles.matrix;

    case 'Hipercubo Latino'
        for i=1:limite2
            lim=limite2-n
            for j=1:limite1
                random=fix(rand*lim+1);
                vrandom(j)=handles.matrix(j,random);
                handles.matrix(j,random)=handles.matrix(j,lim);
            end
            n=n+1
            if (handles.flag2==0)
                handles.matrix_out=vrandom;
                handles.flag2=1;
            else
                handles.matrix_out=[handles.matrix_out ; vrandom];
            end
        end
    end
iterador=1:limite2;
iterador=iterador';
mteste={'Teste de Normalidade' 'a-aceite' 'd-dúvida' 'n-não aplicável' 'r -
rejeitado';'Teste de Kolmogorov-Smirnov' 'valor positivo [0;1,364]-
verifica' '-2 - não verifica' '-1 - não aplicável' ' '};
ficheiro=[get(handles.edtficheiro,'String'),'xlsx'];
xlswrite(ficheiro,handles.vnomes,'Dados_Matlab','B1');
xlswrite(ficheiro,handles.vnomes,'Testes_de_hipótese','B3');
Xlswrite(ficheiro,mteste,'Testes_de_hipótese','B1');
xlswrite(ficheiro,{'Teste Norm'},'Testes_de_hipótese','A4');
xlswrite(ficheiro,handles.vhipotese,'Testes_de_hipótese','B4');
xlswrite(ficheiro,{'K-S'},'Testes_de_hipótese','A5');
xlswrite(ficheiro,handles.vhipoteseks,'Testes_de_hipótese','B5');
xlswrite(ficheiro,handles.matrix_out,'Dados_Matlab','B2');
xlswrite(ficheiro,'N','Dados_Matlab','A1');
xlswrite(ficheiro,iterador,'Dados_Matlab','A2');
xlswrite(ficheiro,handles.seed,'Testes_de_hipótese','B7');
seed={'seed'};
xlswrite(ficheiro,seed,'Testes_de_hipótese','A7');

msg=['Ficheiro ',ficheiro,' gerado com sucesso...'];
msgbox(msg,'Rangen 2.0');

guidata(hObject, handles);

% Declarações dos espaços editáveis do primeiro módulo-----

```

```
function edtficheiro_Callback(hObject, eventdata, handles)
```

```
function edtficheiro_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function pop2_Callback(hObject, eventdata, handles)
```

```
function pop2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function pbmod1_Callback(hObject, eventdata, handles)
```

```
% Botões que definem qual dos módulos está activo-----
```

```
    set(handles.edt1,'Enable','On');
    set(handles.edt2,'Enable','On');
    set(handles.edt3,'Enable','On');
    set(handles.nsamples,'Enable','On');
    set(handles.edtnome,'Enable','On');
    set(handles.txtnr,'Enable','On');
    set(handles.pb1,'Enable','On');
    set(handles.pbgerar,'Enable','On');
    set(handles.popupmenu,'Enable','On');
    set(handles.pop2,'Enable','On');
    set(handles.edtficheiro,'Enable','On');
    set(handles.edtficheiro2,'Enable','Off');
    set(handles.edtfolha1,'Enable','Off');
    set(handles.edtcell1,'Enable','Off');
    set(handles.edtcell2,'Enable','Off');
    set(handles.edtcell3,'Enable','Off');
    set(handles.edtcell4,'Enable','Off');
    set(handles.pbimport1,'Enable','Off');
    set(handles.pbimport2,'Enable','Off');
    set(handles.pbcalc,'Enable','Off');
    set(handles.nsamples2,'Enable','Off');
    set(handles.edtexport,'Enable','Off');
    set(handles.pbexport,'Enable','Off');
    set(handles.poggerar,'Enable','Off');
    set(handles.pbgerar2,'Enable','Off');
    set(handles.edtncolunas,'Enable','Off');
    set(handles.pbexportgraf,'Enable','Off');
    set(handles.pbreset1,'Enable','On');
    set(handles.pbreset2,'Enable','Off');
```

```
function pbmod2_Callback(hObject, eventdata, handles)
```

```
    set(handles.edt1,'Enable','Off');
    set(handles.edt2,'Enable','Off');
    set(handles.edt3,'Enable','Off');
    set(handles.nsamples,'Enable','Off');
    set(handles.edtnome,'Enable','Off');
    set(handles.txtnr,'Enable','Off');
    set(handles.pb1,'Enable','Off');
```

```

set(handles.pbgerar,'Enable','Off');
set(handles.edtficheiro,'Enable','Off');
set(handles.popupmenu,'Enable','Off');
set(handles.pop2,'Enable','Off');
set(handles.edtficheiro2,'Enable','On');
set(handles.edtfolha1,'Enable','On');
set(handles.edtcell1,'Enable','On');
set(handles.edtcell2,'Enable','On');
set(handles.edtcell3,'Enable','On');
set(handles.edtcell4,'Enable','On');
set(handles.pbimport1,'Enable','On');
set(handles.pbimport2,'Enable','On');
set(handles.pbcalc,'Enable','On');
set(handles.nsamples2,'Enable','On');
set(handles.edtexport,'Enable','On');
set(handles.pbexport,'Enable','On');
set(handles.popgerar,'Enable','On');
set(handles.pbgerar2,'Enable','On');
set(handles.edtncolunas,'Enable','On');
set(handles.pbexportgraf,'Enable','On');
set(handles.pbreset1,'Enable','Off');
set(handles.pbreset2,'Enable','On');

```

```
function edtcell1_Callback(hObject, eventdata, handles)
```

```
% Declarações dos espaços editáveis do segundo módulo-----
```

```
function edtcell1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function edtcell2_Callback(hObject, eventdata, handles)
```

```
function edtcell2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function edtcell3_Callback(hObject, eventdata, handles)
```

```
function edtcell3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function edtcell4_Callback(hObject, eventdata, handles)
```

```
function edtcell4_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function edtficheiro2_Callback(hObject, eventdata, handles)
```

```
function edtficheiro2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
```

```

    set(hObject,'BackgroundColor','white');
end

function pbimport1_Callback(hObject, eventdata, handles)
% Botão "Importar" (R)-----Início-----
-----Importação dos coluna que contém o Momento resistente-----

ficheiro2=[get(handles.edtficheiro2,'String'),'.xlsx'];
folha=[get(handles.edtfolha1,'String')];
cell1=[get(handles.edtcell1,'String'),':',get(handles.edtcell2,'String')];
handles.mresistente=xlsread(ficheiro2,folha,cell1);
guidata(hObject, handles);

%-----Teste de Normalidade-----
    flagcond=0;
    hipotese='';
    norm11=0;
    norm12=0;
    norm21=0;
    norm22=0;
    norm1=0;
    norm2=0;
    condi1=0;
    condi2=0;
    condi22=0;
    condi3=0;
    condi44=0;
    desvio1=0;
    desvio2=0;
    limite2=str2double(get(handles.nsamples2,'String'))

    medial=mean(handles.mresistente);
    desvior=std(handles.mresistente);
    matrixresistente=handles.mresistente';
    for i=1:1:limite2
        norm11=norm11+(matrixresistente(i)-medial)^3;
        norm12=norm12+(matrixresistente(i)-medial)^2 ;

        norm21=norm21+(matrixresistente(i)-medial)^4;
        norm22=norm22+(matrixresistente(i)-medial)^2 ;
    end
    norm1=(1/limite2*norm11)/(1/limite2*norm12)^(3/2);
    desvio1=sqrt((6*(limite2-2))/((limite2+1)*(limite2+3)));

    norm2=(1/limite2*norm21)/(1/limite2*norm22)^(2)-3;
    desvio2=sqrt((24*limite2*(limite2-2)*(limite2-3))/((limite2+1)^2*(limite2+3)*(limite2+5)));

    if (abs(norm1)<1.5*desvio1 &&
        (abs(norm2+6/(limite2+1))<1.5*desvio2))
        hipotese='a';
    elseif( (abs(norm1)<2*desvio1 &&
        (abs(norm2+6/(limite2+1))<2*desvio2)))
        hipotese='d';
    else
        hipotese='r';
    end
    guidata(hObject, handles);

```

```

handles.vhipotesemod2(1)=hipotese;
guidata(hObject, handles);
%-----Fim do Teste de Normalidade-----

%-----Teste de Kolmogorov-Sirnov-----
switch hipotese
    case 'a'
        msg=['O sistema Resistente, R, verifica as duas
condições do teste de normalidade, aceita-se a hipótese
de normalidade!'];
        msgbox(msg, 'Rangen 2.0');
        handles.vhipoteseksmod2(1)=-1;
        guidata(hObject, handles);
    case 'r'
        Vks=sort(handles.mresistente);
        Vteorico=Vks';
        for ks=1:limite2
            vksf(ks)=ks/(limite2+1);
            vksf=vksf;
        end
        yteorico=normcdf(Vteorico,medial,desvior);
        Dn=max(abs(yteorico-vksf));

        Dnn=Dn*(sqrt(str2double(get(handles.nsamples2, 'String'))
+0.12+0.11/sqrt(str2double(get(handles.nsamples2, 'String'
))))))
        if Dnn>1.36428
            msg=['O sistema Resistente, R, não verifica o teste de
normalidade e o teste de Kolmogorov-Smirnov!'];
            msgbox(msg, 'Rangen 2.0');
            handles.vhipoteseksmod2(1)= -2;
        else
            msg=['O sistema Resistente, R, não verifica o teste de
normalidade, mas verifica o teste de Kolmogorov-
Smirnov!'];
            msgbox(msg, 'Rangen 2.0');
            handles.vhipoteseksmod2(1)= Dnn;
        end
    case 'd'
        Vks=sort(handles.mresistente);
        Vteorico=Vks';
        for ks=1:limite2
            vksf(ks)=ks/(limite2+1);
            vksf=vksf;
        end
        yteorico=normcdf(Vteorico,medial,desvior);
        Dn=max(abs(yteorico-vksf));
        Dnn=Dn*(sqrt(str2double(get(handles.nsamples2, 'String'))
+0.12+0.11/sqrt(str2double(get(handles.nsamples2, 'String'
))))))
        if Dnn>1.36428
            msg=['O sistema Resistente, R, verifica a segunda
condição do teste de normalidade, mas não verifica o
teste de Kolmogorov-Smirnov!'];
            msgbox(msg, 'Rangen 2.0');
            handles.vhipoteseksmod2(1)= -2;
        else

```

```

        msg=['O sistema Resistente, R, verifica a segunda
        condição do teste de normalidade e verifica o teste de
        Kolmogorov-Smirnov!'];
        msgbox(msg, 'Rangen 2.0');
        handles.vhipoteseksmod2(1)= Dnn;
    end
        guidata(hObject, handles);
    end
guidata(hObject, handles);
%---Fim do Teste de Kolmogorov-Smirnov-----

function pbimport2_Callback(hObject, eventdata, handles)
% Botão "Importar" (S)-----Início-----
----Importação dos coluna que contém o Momento proveniente das acções----

ficheiro2=[get(handles.edtficheiro2, 'String'), '.xlsx'];
folha=[get(handles.edtfolha1, 'String')];
cell2=[get(handles.edtcell13, 'String'), ':', get(handles.edtcell14, 'String')];
handles.msolicitador=xlsread(ficheiro2, folha, cell2);
guidata(hObject, handles);

%-----Teste de Normalidade-----

flagcond=0;
hipotese='';
norm11=0;
norm12=0;
norm21=0;
norm22=0;
norm1=0;
norm2=0;
condi1=0;
condi2=0;
condi22=0;
condi3=0;
condi44=0;
desvio1=0;
desvio2=0;
limite2=str2double(get(handles.nsamples2, 'String'));

medial=mean(handles.msolicitador);
desvior=std(handles.msolicitador);
%fprintf('matrix= %d\n', handles.msolicitador);
matrixsolicitador=handles.msolicitador';
%fprintf('matrixsoli= %d\n', matrixsolicitador);

for i=1:1:limite2
    norm11=norm11+(matrixsolicitador(i)-medial)^3;
    norm12=norm12+(matrixsolicitador(i)-medial)^2;

    norm21=norm21+(matrixsolicitador(i)-medial)^4;
    norm22=norm22+(matrixsolicitador(i)-medial)^2 ;
end
norm1=(1/limite2*norm11)/(1/limite2*norm12)^(3/2);
desvio1=sqrt((6*(limite2-2))/((limite2+1)*(limite2+3)));

norm2=(1/limite2*norm21)/(1/limite2*norm22)^(2)-3;

```

```

desvio2=sqrt((24*limite2*(limite2-2)*(limite2-3))/((limite2+1)^2*(limite2+3)*(limite2+5)));
if (abs(norm1)<1.5*desvio1 &&
(abs(norm2+6/(limite2+1))<1.5*desvio2))
    hipotese='a';
elseif( (abs(norm1)<2*desvio1 &&
(abs(norm2+6/(limite2+1))<2*desvio2)))
    hipotese='d';
else
    hipotese='r';
end
guidata(hObject, handles);

handles.vhipotesemod2(2)=hipotese;

guidata(hObject, handles);

```

```

%-----Fim do Teste de Normalidade-----

```

```

%-----Teste de Kolmogorov-Sirnov-----

```

```

switch hipotese
    case 'a'
        msg=['O vector Acção solicitada, S, verifica as duas condições
do teste de normalidade, aceita-se a hipótese de
normalidade!'];
        msgbox(msg, 'Rangen 2.0');
        handles.vhipoteseksmod2(2)=-1;
        guidata(hObject, handles);
    case 'r'
        Vks=sort(handles.msolicitador);
        Vteorico=Vks';
        for ks=1:limite2
            vksf(ks)=ks/(limite2+1);
            vksf=vksf;
        end
        yteorico=normcdf(Vteorico,medial,desvior);
        Dn=max(abs(yteorico-vksf));
        Dnn=Dn*(sqrt(str2double(get(handles.nsamples2, 'String')))+0.12+
0.11/sqrt(str2double(get(handles.nsamples2, 'String'))))
        if Dnn>1.36428
            msg=['O vector da Acção solicitada, S, não verifica o teste
de normalidade e o teste de Kolmogorov-Smirnov!'];
            msgbox(msg, 'Rangen 2.0');
            handles.vhipoteseksmod2(2)= -2;
        else
            msg=['O vector da Acção solicitada, S, não verifica o teste de
normalidade, mas verifica o teste de Kolmogorov-Smirnov!'];
            msgbox(msg, 'Rangen 2.0');
            handles.vhipoteseksmod2(2)= Dnn;
        end
    case 'd'
        Vks=sort(handles.msolicitador);
        Vteorico=Vks';
        for ks=1:limite2
            vksf(ks)=ks/(limite2+1);
            vksf=vksf;
        end
        yteorico=normcdf(Vteorico,medial,desvior);

```

```

        Dn=max(abs(yteorico-vksf));
        Dnn=Dn*(sqrt(str2double(get(handles.nsamples2,'String'))
        +0.12+0.11/sqrt(str2double(get(handles.nsamples2,'String'
        ))))
    if Dnn>1.36428
        msg=['O vector da Acção solicitada, S, verifica a segunda
        condição do teste de normalidade, mas não verifica o
        teste de Kolmogorov-Smirnov!'];
        msgbox(msg,'Rangen 2.0');
        handles.vhipoteseksmod2(2)= -2;
    else
        msg=['O vector da Acção solicitada, S, verifica a segunda
        condição do teste de normalidade e verifica o teste de
        Kolmogorov-Smirnov!'];
        msgbox(msg,'Rangen 2.0');
        handles.vhipoteseksmod2(2)= Dnn;
    end
    guidata(hObject, handles);
end

%---Fim do Teste de Kolmogorov-Smirnov-----

guidata(hObject, handles);

function edtfolhal_Callback(hObject, eventdata, handles)
% Declarações dos espaços editáveis do segundo módulo-----

function edtfolhal_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pbcalc_Callback(hObject, eventdata, handles)
% Botão "Calcular"-----Início-----
Cálcula o índice de fiabilidade, a probabilidade de rotura, o erro e o COV-

edt11=str2double(get(handles.nsamples2,'String'));
edt11=edt11-edt11;
if edt11==0
else
    errordlg('Dados incompatíveis!!! Altere os dados introduzidos no espaço
    "N".')
    dbstop
end
ur=mean(handles.mresistente);
vr=std(handles.mresistente);
us=mean(handles.msolicitador);
vs=std(handles.msolicitador);

mr=handles.mresistente;
ms=handles.msolicitador;
handles.msistema=mr-ms;
n=str2double(get(handles.nsamples2,'String'))
j=0;
for i=1:1:n

```

```

        if handles.msistema(i)<0
            j=j+1;
        else
            j=j;
        end
    end

    pff=j/n
    handles.uz=ur-us;
    handles.vz=sqrt(vr^2+vs^2);

    beta=((ur-us)/sqrt(vr^2+vs^2));
    pf=normcdf(-beta,0,1);
    cov=(sqrt((1-pf)*pff/n)/pff)
    erro=(sqrt((1-pf)/(n*pf))*200)
    handles.vresultados={pf,beta,cov,erro};
    msg=['Cálculo efectuado com sucesso...'];
    msgbox(msg,'Rangen 2.0');
    guidata(hObject, handles);

function nsamples2_Callback(hObject, eventdata, handles)
% Declarações dos espaços editáveis do segundo módulo-----

function nsamples2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pbexport_Callback(hObject, eventdata, handles)
% Botão "Exportar" -----Início-----
----Exportação dos Resultados para Excel-----

mediadesvio=[handles.uz handles.vz];
handles.vnomesfinal={'Pf','Beta','COV','Erro(%)'};
ficheiro=[get(handles.edtexport,'String'),'xlsx'];
xlswrite(ficheiro,handles.vnomesfinal,'Resultados_Matlab','A1');
xlswrite(ficheiro,handles.vresultados,'Resultados_Matlab','A2');
xlswrite(ficheiro,'RS','Resultados_Matlab','B4');
xlswrite(ficheiro,{'Teste de norm';'K-S';'Z'},'Resultados_Matlab','A4');
xlswrite(ficheiro,{'uz','vz'},'Resultados_Matlab','B6');
xlswrite(ficheiro,(mediadesvio),'Resultados_Matlab','b7');
%xlswrite(ficheiro,{'K-S'},'Resultados_Matlab','A5');
xlswrite(ficheiro,handles.vhipotesemod2,'Resultados_Matlab','B4');
xlswrite(ficheiro,handles.vhipoteseksmod2,'Resultados_Matlab','B5');
xlswrite(ficheiro,handles.msistema,'Resultados_Matlab','A7');
msg=['Ficheiro ',ficheiro,' exportado com sucesso...'];
msgbox(msg,'Rangen 2.0');
guidata(hObject, handles);

function edtexport_Callback(hObject, eventdata, handles)
% Declarações dos espaços editáveis do segundo módulo-----

function edtexport_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

```

```
        set(hObject,'BackgroundColor','white');
    end

function popgerar_Callback(hObject, eventdata, handles)

function popgerar_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pbgerar2_Callback(hObject, eventdata, handles)
% Botão "Gerar" (Gráfico)-----Início-----
%-----Geração dos histogramas nos eixos-----

n=str2double(get(handles.edtncolunas,'String'));
string = get(handles.popgerar, 'String');
value = get(handles.popgerar, 'Value');

switch string{value};
    case 'Resistente'
        handles.axes2;
        histfit(handles.mresistente,n)
    case 'Acção'
        handles.axes2;
        histfit(handles.msolicitador,n);
    case 'Sistema estrutural'
        handles.axes2;
        histfit(handles.msistema,n);
end
function edtncolunas_Callback(hObject, eventdata, handles)
% Declarações dos espaços editáveis do segundo módulo-----

function edtncolunas_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pbexportgraf_Callback(hObject, eventdata, handles)
% Botão "Exportar gráfico"-----Início-----
%-----Geração dos histogramas numa janela destacada-----

n=str2double(get(handles.edtncolunas,'String'));
string = get(handles.popgerar, 'String');
value = get(handles.popgerar, 'Value');

switch string{value};
    case 'Resistente'
        figure;
        histfit(handles.mresistente,n)

    case 'Acção'
        figure;
        histfit(handles.msolicitador,n);
    case 'Sistema estrutural'
        figure;
```

```

histfit(handles.msistema,n);

end

function pbreset1_Callback(hObject, eventdata, handles)
% Botão "Reset" -----
--Primeiro e segundo Módulo-----

handles.matrix=0;
handles.matrix_iterator=1;
handles.flag1= 0;
handles.flag2= 0;
handles.matrix_out=0;
handles.vnomes='';
handles.vresultados=0;
handles.vhipotese='';
handles.vhipoteseks=0;
handles.vksererror=0;

set(handles.edt1,'String','');
set(handles.edt2,'String','');
set(handles.edt3,'String','');
set(handles.nsamples,'String','');
set(handles.edtnome,'String','');
set(handles.txtnr,'String','0');
set(handles.edtficheiro,'String','default');
set(handles.nsamples,'Enable','On');
set(handles.pop2,'Enable','On');
set(handles.popupmenu,'Enable','On');
set(handles.pbgerar,'Enable','On');
set(handles.pb1,'Enable','On');
clc;
guidata(hObject, handles);

function pbreset2_Callback(hObject, eventdata, handles)
handles.mresistente=0;
handles.msolicitador=0;
handles.msistema=0;
handles.vnomesfinal='';
handles.uz=0;
handles.vz=0;
handles.vhipotesemod2=['e' 'e'];
handles.vhipoteseksmod2=[0 0];
set(handles.edtexport,'String','default');
set(handles.edtficheiro2,'String','default');
set(handles.edtfolha1,'String','Dados_Matlab');
set(handles.edtcell1,'String','');
set(handles.edtcell2,'String','');
set(handles.edtcell3,'String','');
set(handles.edtcell4,'String','');
set(handles.nsamples2,'String','');
set(handles.edtncolunas,'String','');
cla;
clc;
guidata(hObject, handles);

function edtseed_Callback(hObject, eventdata, handles)

```

```
function edtseed_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
%Fim do programa de apoio à avaliação de fiabilidade estrutural-----
```