



XII CONGRESSO BRASILEIRO
de PONTES e ESTRUTURAS
7 a 11 de junho de 2021 - Congresso Virtual

Comparação do Método Form e Monte Carlo Sequenciais e Paralelizados Numa Análise De Confiabilidade da Função Estado Limite Explícita (Viga em Plastificação)

Wilson Rodríguez Calderón¹, Myriam Rocío Pallares Muñoz²

¹Universidad Santo Tomás / Posgrados / Tunja, Colombia. Email: wilroca50@hotmail.com

²Universidad Surcolombiana / Neiva, Colombia. / Email: myriam.pallares@usco.edu.co

Resumo

É apresentado o desenvolvimento, aplicação e comparação de códigos informáticos em Matlab para o cálculo da confiabilidade estrutural numa Viga sujeita a plastificação. A função de estado limite é explicitamente proposta com base na formulação de Ang e Tang (ANG E TANG, 1984). Códigos computacionais são desenvolvidos para o Método de Confiabilidade de Primeira Ordem (FORM) e o método Monte Carlo (MC) (implementação de recuperação de simulações anteriores, variáveis antitéticas, paralelização de CPU e GPU). Os códigos computacionais obtidos são aplicados ao caso de uma Viga submetida a plastificação de flexão (variáveis aleatórias normais e não correlacionadas) e obtém-se as probabilidades de falha, índices de confiabilidade e tempos de CPU que permitem estabelecer o desempenho numérico e de custo dos diferentes métodos e suas implementações, incluindo as versões paralelas. Conclusões importantes são obtidas quanto aos resultados de confiabilidade estrutural encontrados com os diferentes métodos abordados, os custos computacionais no tempo de CPU, as vantagens e desvantagens dos métodos de confiabilidade estudados e finalmente uma avaliação do desempenho das paralelizações desenvolvidas para CPU e GPU ou placas gráficas. Um quadro teórico envolvendo conceitos associados aos métodos utilizados é apresentado na primeira parte do artigo.

Palavras-chave

Confiabilidade Estrutural, Paralelização CPU e GPU, Método FORM, Método Monte Carlo, Variáveis antitéticas.

INTRODUÇÃO

O trabalho realizado baseia-se no problema de cálculo da confiabilidade estrutural numa viga submetida à plastificação., esse problema foi proposto por Ang e Tang (ANG E TANG, 1984) ao formular a função de estado-limite explícito que caracteriza o fenômeno, portanto é um teste de comparação muito bom (Benchmark) para o estudo do desempenho de diferentes métodos de cálculo de confiabilidade e sua implementação computacional. Procura-se explorar a eficiência computacional dos métodos de confiabilidade de primeira ordem (FORM) e Monte Carlo Crudo (MC) com algumas melhorias (variáveis antitéticas), bem como técnicas de paralelização em CPU e GPU.

Dadas as altas necessidades de cálculo que surgem naturalmente em questões de confiabilidade estrutural, procura-se alternativas de paralelização que diminuam os tempos de cálculo, seja paralelizando o código na CPU para aproveitar os núcleos do processador (2, 4, 6, 8, ...) ou por computação acelerada em uma unidade de processamento gráfico (GPU) onde temos centenas ou milhares de núcleos.

Atualmente, os aceleradores de GPU estão fortemente se dedicando à aceleração de aplicações de plataformas de Inteligência artificial (IA), automóveis, robôs, drones, mineração de dados, cálculo numérico, software CAD e CAE, entre outros. A Figura 1 mostra um esquema de computação acelerado por GPU.

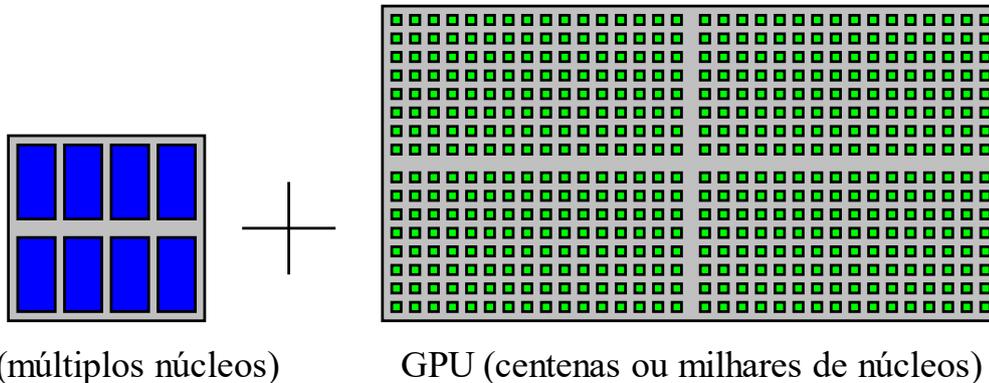


Figura 1 – Configuração da computação acelerada por GPU.

ASPECTOS TEÓRICOS DOS MÉTODOS PARA CÁLCULO DA CONFIABILIDADE EM PROBLEMAS ESTRUTURAIS

Abaixo está um resumo dos aspectos teóricos dos métodos ou algoritmos usados como são:

- 1) Método de confiabilidade de primeira ordem FORM versão HRLF (“First Order Reliability Methods”).
- 2) Método de simulação de Monte Carlo.
- 3) Estratégia de variáveis antitéticas como técnica de redução de variância.

Método de confiabilidade de primeira ordem FORM versão HRLF (“First Order Reliability Methods”)

O método FORM aparece para quando as variáveis aleatórias U (espaço real ou original U), podem ter qualquer distribuição probabilística, podem ser correlacionadas ou não. Si as variáveis estiverem correlacionadas, são necessários processos de transformação (ROSENBLATT, 1952; NATAF, 1962) para chegar à descrição em variáveis normais padrões reduzidas e não correlacionadas (espaço normal padrão \mathbf{u}). A função de estado limite $g(U)$ no espaço real pode ser exibida no espaço reduzido \mathbf{u} em função das variáveis \mathbf{u} como $g(\mathbf{u})$. A superfície de falha $g(\mathbf{u}) = 0$ é aproximada por um hiperplano no ponto com a menor distância até a origem, identificado como o ponto de projeto (“Most Probable Point”–MPP) \mathbf{u}^* , lembrando que ele é também o ponto sobre o hiperplano, com o maior valor da função densidade de probabilidade conjunta das variáveis. Encontrado o ponto de projeto, pode calcular-se o índice de confiabilidade (β) que é a distância deste ponto de projeto até a origem pela Equação (1) e a probabilidade de falha (P_f) pode ser calculada aproximadamente pela Equação (2). Na Figura 2 é possível ver o processo para duas variáveis aleatórias.

$$\beta = \sqrt{\mathbf{u}^{*T} \mathbf{u}^*} \quad (1)$$

$$P_f = P(g(\mathbf{u}) < 0) \cong \Phi(-\beta) \quad (2)$$

onde, Φ é a função cumulativa de probabilidades (FCP) normal padrão no espaço reduzido e \mathbf{u}^* é o ponto de projeto no espaço reduzido.

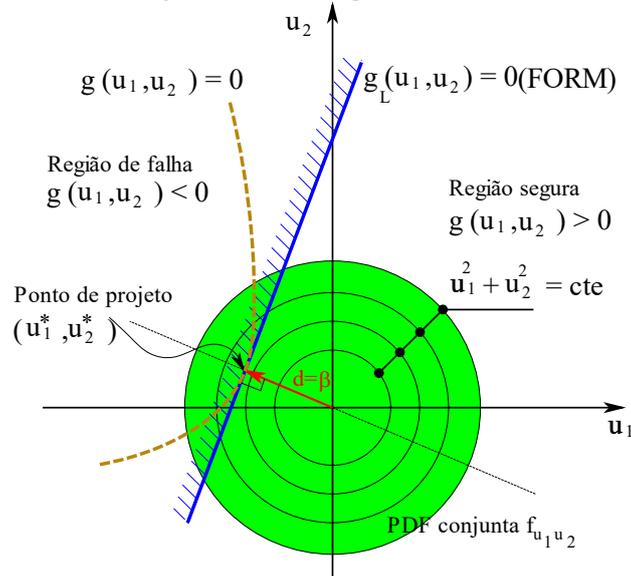


Figura 2 – Descrição das incógnitas para a determinação do ponto de projeto no método FORM para duas variáveis aleatórias.

A determinação do ponto de projeto \mathbf{u}^* da Figura 2 é a base para calcular o índice de confiabilidade e a probabilidade de falha no método FORM. O método FORM encontra o ponto de projeto pela abordagem de um problema de otimização que mais exatamente consiste num problema de programação não-linear com uma restrição, assim:

$$\begin{aligned} & \text{minimizar } \|\mathbf{u}\| \\ & \text{sujeito a: } g(\mathbf{u}) = 0 \end{aligned} \quad (3)$$

O problema apresentado na Equação (3) pode ser resolvido usando a teoria dos métodos de otimização disponíveis e adaptados ao problema de otimização por diferentes alternativas. O esquema mais usado para problemas de análise de confiabilidade em geral é o desenvolvido por Hasofer e Lind (HASOFER E LIND, 1974) e aperfeiçoado por Rackwitz e Fiessler (RACKWITZ E FIESSLER, 1978), na literatura encontra-se habitualmente identificado como HLRF e está expresso recursivamente por:

$$\mathbf{u}^{k+1} = \frac{1}{\|\nabla g(\mathbf{u}^k)\|^2} [\nabla g(\mathbf{u}^k)^T \mathbf{u}^k - g(\mathbf{u}^k)] \nabla g(\mathbf{u}^k) \quad (4)$$

onde $g(\mathbf{u}^k)$ é o valor da função de falha avaliado no ponto \mathbf{u} na iteração k e $\nabla g(\mathbf{u}^k)$ é o gradiente da função de falha no espaço reduzido avaliado no ponto \mathbf{u} na iteração k .

Método de simulação de Monte Carlo (MC)

O método de Monte Carlo (MC) é um método de cálculo de probabilidade baseado no princípio de simulações aleatórias, apareceu no ano de 1949, com a publicação dos autores matemáticos Nicolás Metrópolis e Stanislaw Ulam (METRÓPOLIS E ULAM, 1949). Este método apresenta facilidade para a implementação computacional serial e paralela, boa precisão, verificação estatística do erro e não precisa maiores conhecimentos matemáticos.

À medida que os problemas se tornam complexos, a simulação aparece como uma boa opção para estimar a confiabilidade. Caso se considere um sistema ou estrutura cujo desempenho é definido por um conjunto de variáveis aleatórias $\mathbf{U} = \{U_1, U_2, \dots, U_n\}$, a função de estado limite (f.e.l) $g(\mathbf{U})$ que define as regiões de falha e segurança e a função de densidade de probabilidade conjunta das variáveis aleatórias $f_U(\mathbf{U}) = f_U(U_1, U_2, \dots, U_n)$. No caso que a função $g(\mathbf{U}) > 0$ indica que \mathbf{U} pertence a região segura (D_s), caso contrário se define a região de falha (D_f) e se $g(\mathbf{U}) = 0$ o estado é de falha iminente, como pode-se ver na Figura 3. Em geral, a probabilidade de falha (P_f) de uma estrutura ou sistema pode ser calculada como:

$$P_f = \int_{g(\mathbf{U}) < 0} f_U(\mathbf{U}) d\mathbf{U} = \int_{D_f} f_U(\mathbf{U}) d\mathbf{U} \quad (5)$$

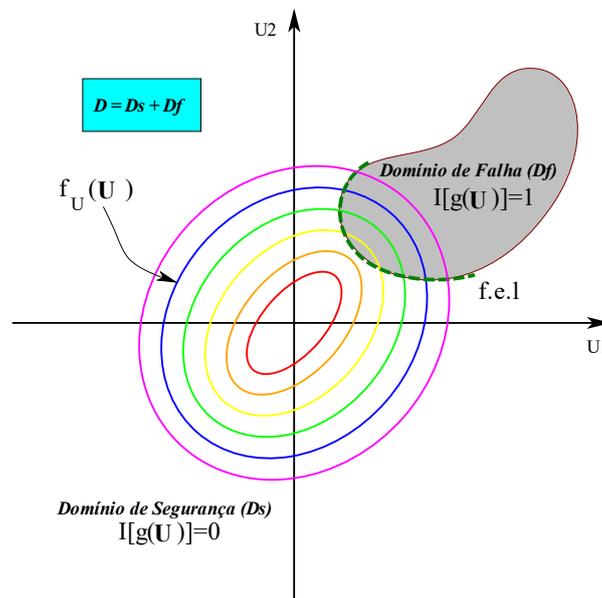


Figura 3 – Descrição dos elementos básicos para o cálculo de probabilidade de falha pelo método de Monte Carlo.

Pode-se definir uma função indicadora I tal que $I[g(\mathbf{U})] = 1$ para $g(\mathbf{U}) < 0$ (Domínio de falha - D_f) e $I[g(\mathbf{U})] = 0$ para $g(\mathbf{U}) > 0$ (Domínio de segurança - D_s). Então, a probabilidade de falha pode ser estimada como o valor esperado da função indicadora. A função indicadora identifica o domínio da integração e a função $g(\mathbf{U})$ é uma função de desempenho na análise de confiabilidade, portanto, a integração fornece a probabilidade de falha, P_f . Observa-se que a integração é realizada em todos os domínios onde as variáveis básicas são definidas (Domínio $D=D_s+D_f$). Quando esta integração é avaliada teoricamente pelo método MC para infinitos pontos, a P_f pode ser calculada da seguinte maneira:

$$P_f = E\{I[g(\mathbf{U})]\} = \mu_I = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N I[g(\mathbf{U}_i)] \quad (6)$$

onde N é o número de simulações. Na prática é possível calcular um estimador de P_f (usa-se a notação \hat{P}_f para isso estimador) pelo método MC, assim:



XII CONGRESSO BRASILEIRO
de PONTES e ESTRUTURAS
7 a 11 de junho de 2021 - Congresso Virtual

$$\hat{P}_f = \frac{1}{N} \sum_{i=1}^N I[g(\mathbf{U}_i)] \quad (7)$$

Lembrando a definição do coeficiente de variação aplicado a \hat{P}_f , pode ser demonstrado que (GOMES, 2004):

$$CV_{\hat{P}_f} = \frac{\sigma_{\hat{P}_f}}{\mu_{\hat{P}_f}} = \frac{\sqrt{\frac{P_f - P_f^2}{N}}}{P_f} = \sqrt{\frac{1 - P_f}{NP_f}} \quad (8)$$

Estratégia de variáveis antitéticas como técnica de redução de variância

O método das variáveis antitéticas (HAMMERSLEY E MORTON, 1956) é uma técnica de simulação para garantir a correlação negativa entre duas amostras de variáveis aleatórias. No método, uma correlação negativa é induzida entre os diferentes ciclos de simulação para diminuir a variância do valor médio estimado. Se U é uma variável aleatória uniformemente distribuída entre 0 e 1 usada na primeira execução de geração de uma variável aleatória básica X_i , então outra variável aleatória $1 - U$ pode ser introduzida em a segunda geração. Se U é uniformemente distribuído entre 0 e 1, então $1 - U$ também é uniformemente distribuído no mesmo intervalo e é negativamente correlacionado com U (HALDAR E MAHADEVAN, 2000).

DESCRIÇÃO DO PROBLEMA

A Capacidade de plastificação por flexão de uma viga de aço pode ser expressa como YZ , onde:

$$\begin{aligned} X_1 &= Y = \text{Resistência à Fluência do aço.} \\ X_2 &= Z = \text{Módulo da seção transversal.} \end{aligned} \quad (9)$$

Se o momento de flexão aplicado a seção relevante é M (X_3), a função de estado limite pode ser definida como:

$$g(X) = X_1 X_2 - X_3 = YZ - M \quad (10)$$

Assumindo que as variáveis são não correlacionadas, determine a confiabilidade da viga.

Suponha-se uma viga com $\bar{Y} = 40$ ksi e $\bar{Z} = 50$ in³ que está sujeita a um momento de flexão $\bar{M} = 1000$ in-kips; e os correspondentes coeficientes de variação COV são:

$$\Omega_Y = 0.125 ; \Omega_Z = 0.05 ; \Omega_M = 0.20 \quad (11)$$

Os desvios padrão correspondentes são então:

$$\begin{aligned} \sigma_Y &= 0.125 * 40 = 5.0 \text{ ksi} \\ \sigma_Z &= 0.05 * 50 = 2.5 \text{ in}^3 \\ \sigma_M &= 0.20 * 1000 = 200 \text{ in} - \text{kip} \end{aligned} \quad (12)$$

A Figura 4 mostra um esquema resumo do problema a ser resolvido.

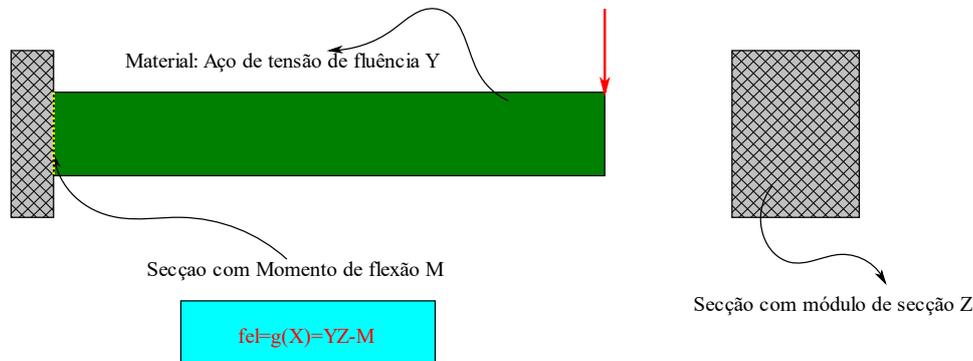


Figura 4 – Esquema do Problema de Confiabilidade.

RESULTADOS E DISCUSSÃO

Abaixo está a solução obtida com o método FORM e o método Monte Carlo. Mostra-se uma comparação das soluções obtidas com os métodos acima e diferentes estratégias de programação sequencial e paralela sobre CPU e GPU.

Solução usando o método FORM

Os resultados obtidos para iteração final são mostrados na Figura 5, obtida diretamente do arquivo de saída (nenhum tipo de regra de ortografia é usado para evitar conflito de caracteres). A tolerância especificada foi de $1E-12$.

```

iteração( 13 )
dGDxnr Variavel( 1 )= 241.542
dGDxnr Variavel( 2 )= 71.376
dGDxnr Variavel( 3 )= -200.000
Xnr Variavel( 1 )= -2.28993
Xnr Variavel( 2 )= -0.67668
Xnr Variavel( 3 )= 1.89610
beta( 13 )= 3.049073476726125
Pf( 13 )= 0.001147741655941
Fo( 13 )= 0.0000000000000000
rk( 13 )= 0.0000000000000000
Tempo CPU do calculo do Metodo Form = 0.05417 s
  
```

Figura 5 – Resultados do Problema de Confiabilidade usando o método FORM.

A Figura 6 mostra o gráfico de convergência do método implementado sendo esta linear (aplicação de conceitos de métodos iterativos, onde a convergência é plotada de maneira semilogarítmica).



XII CONGRESSO BRASILEIRO
de PONTES e ESTRUTURAS
7 a 11 de junho de 2021 - Congresso Virtual

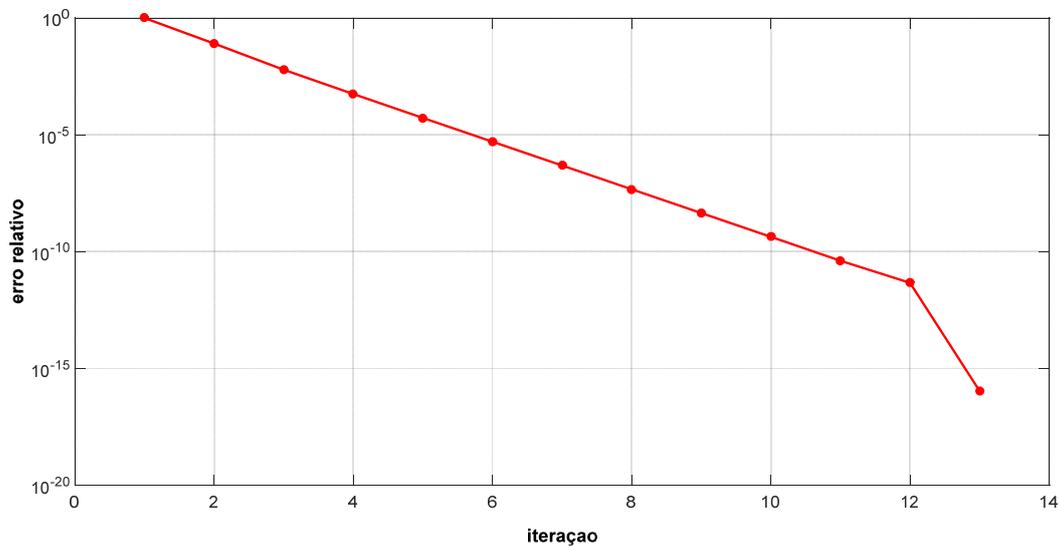


Figura 6 – Curva de Convergência do método FORM.

Solução usando o método de Monte Carlo

A Figura 7 mostra as variáveis aleatórias geradas e a avaliação da função de estado limite, pode ser visto como os valores oscilam em torno de suas respectivas médias.

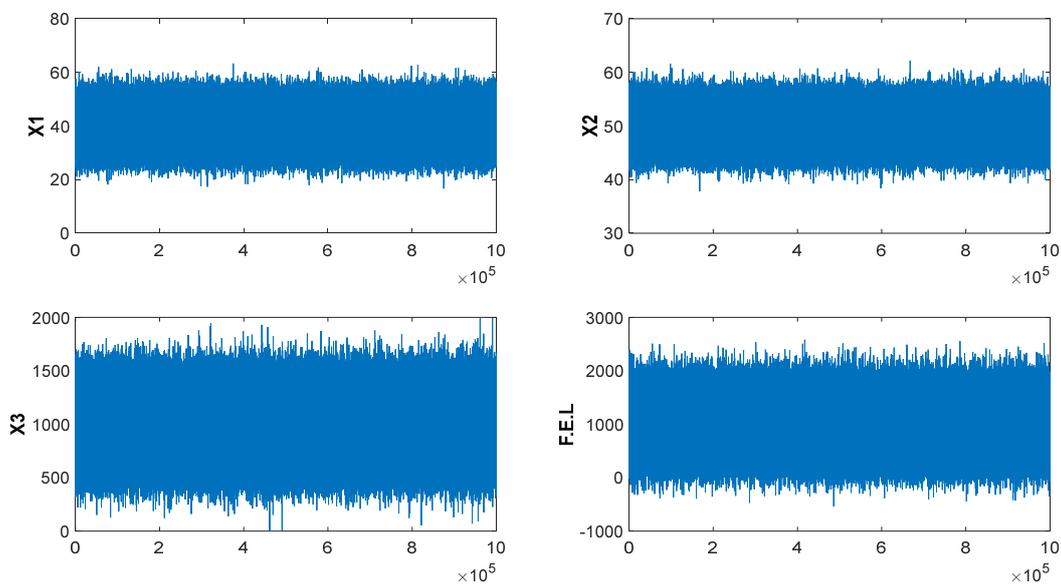


Figura 7 – Variáveis aleatórias geradas e avaliação da função de estado limite no Método Monte Carlo.

A Figura 8 mostra o histograma dos valores da função do estado limite, que finalmente se ajusta a uma distribuição gaussiana normal.

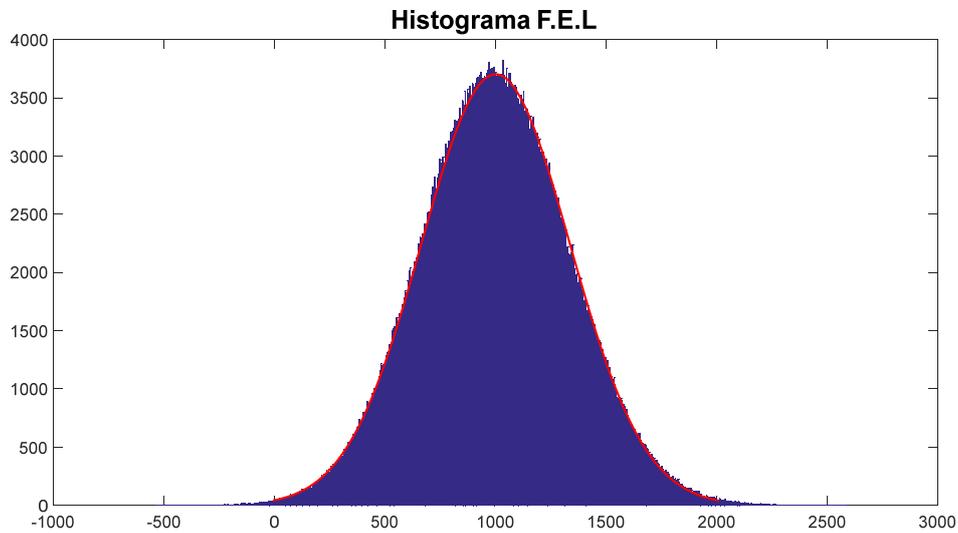


Figura 8 – Histograma da função de estado limite.

A Figura 9 mostra a nuvem de pontos gerada, observando que este conjunto forma uma distribuição espacial de tendência esférica.

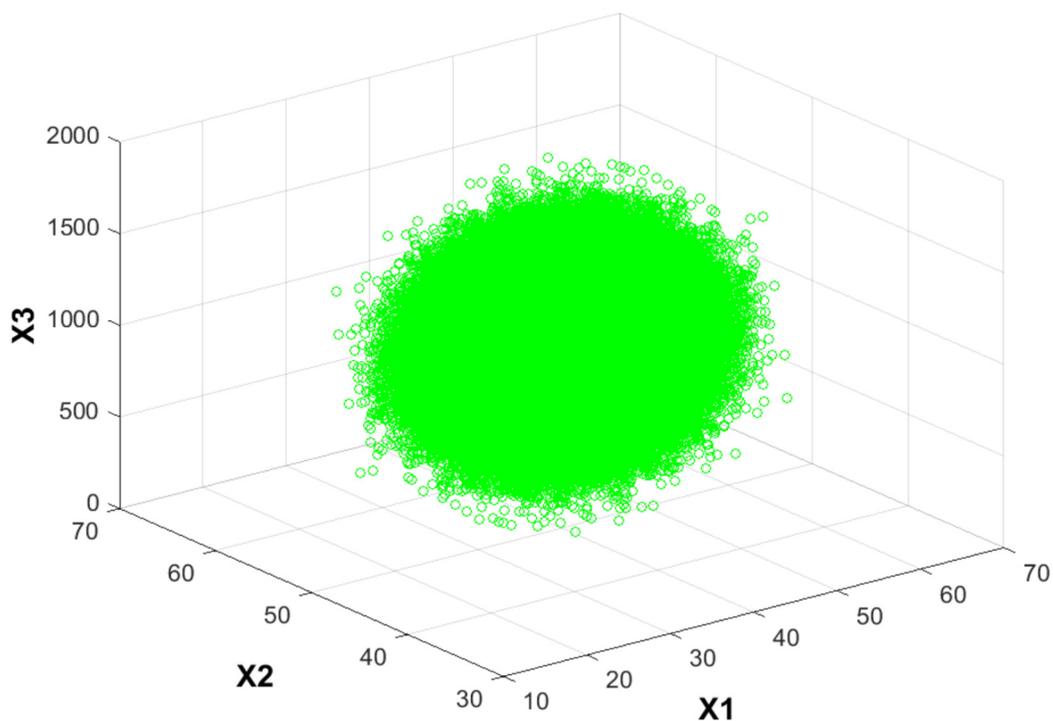


Figura 9 – Nuvem de pontos aleatórios gerados.

Os resultados obtidos são mostrados no listagem da Figura 10, obtida diretamente do arquivo de saída (nenhum tipo de regra de ortografia é usado para evitar conflito de caracteres).

numero de falhas = 1183
 numero de simulacoes = 1000000
 Pf = 0.001183000000000
 CVPf = 0.029056987758820
 beta = 3.039972183123013
 tempo CPU = 49.104690556630629 s

Figura 10 – Resultados do Problema de Confiabilidade usando o método Monte Carlo.

Comparação dos métodos usados e Análise de tempo computacional

A tabela 1 resume os resultados obtidos em códigos dos métodos FORM e Monte Carlo, onde a primeira letra maiúscula indica que ele é o método de Monte Carlo, a segunda letra indica se o código funciona em paralelo ou seqüencial, a opção - r indica que é um código de etapas delineadas que vai rápido, omitindo etapas não necessárias para o problema específico Tratado, a opção -VA indica que o código implementou a estratégia de variáveis antitéticas e finalmente -GPU indica que o código é paralelizado para trabalhar na placa gráfica GPU (NVIDIA - GEFORCE 840). Emprega-se um computador i7, com 2-cores , com 8 GB de RAM e a implementação é feita no programa Matlab. O número de simulações para o método Monte Carlo é 1'000.000 e a tolerância especificada no método FORM é 1E-12.

Tabela 1 – Resultados e tempos computacionais com diferentes códigos em Matlab.

Código	Tempo computacional (s)	Beta	CVpf
Form	0.054	3.049073477	---
Mp-VA	49.105	3.039972183	0.02905699
Ms-VA	218.701	3.026015011	0.02839194
Mp	47.117	3.043298620	0.02921816
Mp-r	12.526	3.045880529	0.02934398
Mp-GPU	859.573	3.046139840	0.02942037

Pode observar-se que o código FORM é o mais rápido, no entanto, o valor mais confiável seria alcançado na implementação das variáveis antitéticas. Pode-se ver que o código seqüencial gasta um pouco mais de 4 vezes o tempo de código paralelo. Por outro lado, pode-se ver que o código paralelo resumido para o caso específico reduz o tempo de execução a apenas 13 segundos aproximados sobre 1'000.000 de simulações que realmente é razoável e competi em economia de tempo contra FORM com as diferenças substanciais entre os dois métodos. A paralelização em GPUs não produziu os resultados esperados e isso pode ter duas opções de argumento para explicar o aumento de tempo, um pode ser que a quantidade de dados não é grande o suficiente para ocupar em calculo aos núcleos e os gastos em comunicação som altos em relação ao trabalho efetivo destes, e o outro pode ser que o código ainda carece de mais funções disponíveis no Matlab para acelerar o processo, no entanto, permanece como trabalho futuro melhorar o código, se é possível investigar mais opções como a avaliação de funções na GPU, por exemplo.

A Figura 11 mostra os dados da tabela 1 brevemente.



XII CONGRESSO BRASILEIRO
de PONTES e ESTRUTURAS
7 a 11 de junho de 2021 - Congresso Virtual

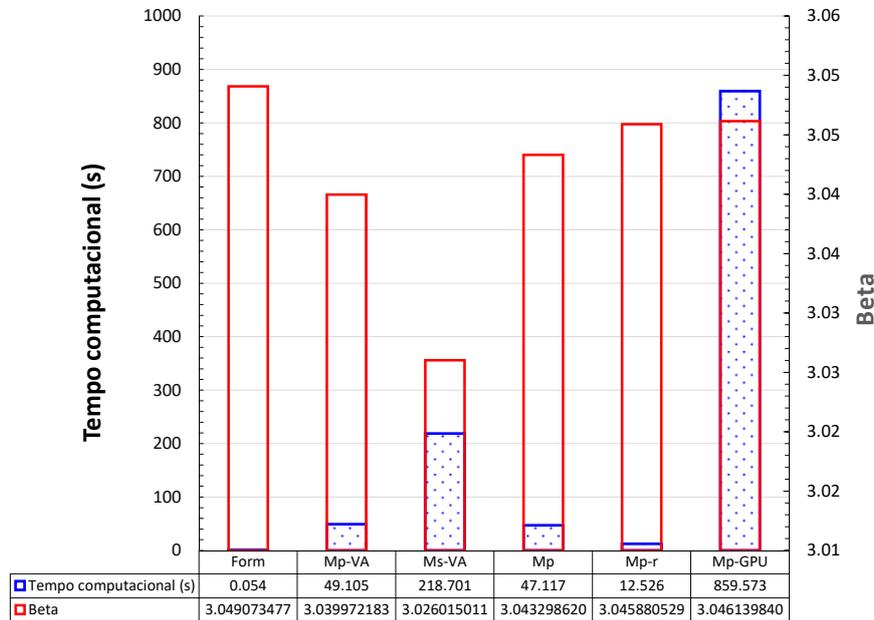


Figura 11 – Gráfico de resultados de índice de confiabilidade e tempo computacional com códigos diferentes.

Conclusões

- Atualmente, a paralelização de códigos mostra ser uma alternativa viável para abordar os elevados custos computacionais verificado em um pequeno exercício desenvolvido (método Monte Carlo), porém com problemas de grande escala se torna uma necessidade, já que a avaliação da função de estado limite não é analítica e em casos usuais em engenharia pode vir de um modelo numérico robusto.
- Uma opção para aproveitar os códigos implementados pode ser tomar FORM como um método de aproximação inicial e o método de Monte Carlo, em segunda instância, como uma alternativa à refinação confiável graças à sua robustez.

Referências

- ANG. A., TANG W., Probability Concepts in Engineering Planning and Design, Vol. I and II, John Wiley & Sons, 1984.
- ROSENBLATT M., Remarks on a multivariate transformation. Ann. Math. Stat., v. 23, p. 470–472, 1952.
- NATAF A., Determination des Distributions dont les Marges sont Donnees. Comptes Rendus de l'Academie des Sciences, v. 225, p. 42–43, 1962.
- HASOFER A. M., LIND N. C., Exact invariant Second-Moment code format, J. Engineering Mechanics Div. ASCE, v.100, n.1, p. 111-121, 1974.
- RACKWITZ R., FIESSLER B., Structural reliability under combined random load sequences, Computers & Structure, vol. 9, pp. 489–494, 1978.
- METROPOLIS N., ULAM S., The Monte Carlo method, Journal of the American Statistical Association 44, 335–341, 1949.
- GOMES H. M., Apostila e slides da disciplina Confiabilidade em Sistemas Mecânicos, UFRGS, 2004.
- HAMMERSLEY J. M., K. W. MORTON, A new Monte Carlo technique antithetic variates, Proc. Camb. Phil. Soc., Vol. 52, pp. 449–474, Variance reduction in antithetic variables, 1956.
- HALDAR A., MAHADEVAN S., Probability, Reliability, and Statistical Methods in Engineering, John Wiley & Sons, 2000.