CONGRESSO DE PONTES E ESTRUTURAS
RIO DE JANEIRO 10 A 12 DE OUTUBRO DE 2005

ENVOLTÓRIA DE ESFORÇOS INTERNOS DEVIDO A AÇÃO DE TRENS-TIPO EM PONTES USANDO ESTRATÉGIA EVOLUTIVA

Gisele Cristina da Cunha Holtz
gisele@tecgraf.puc-rio.br

Luiz Fernando Campos Ramos Martha
lfm@tecgraf.puc-rio.br

Luiz Eloy Vaz
eloyvaz@terra.com.br

Resumo: O objetivo deste trabalho é descrever o procedimento para obtenção de envoltórias de esforços internos devido a cargas móveis. A determinação de valores limites de um esforço em uma seção constitui um problema de otimização cujo objetivo é minimizar ou maximizar os valores dos esforços em relação à posição do trem-tipo que percorre a estrutura. Porém, não existe uma expressão analítica que descreva a variação de um determinado esforço em função da posição do trem-tipo, o que impossibilita o uso da maioria dos métodos clássicos de otimização, porque esses métodos requerem, na maioria das vezes, o uso de pelo menos a primeira derivada da função objetivo em relação às variáveis de projeto. Portanto, este trabalho adotou algoritmos da Estratégia Evolutiva (EE), que é um método probabilístico. Além disso, foi desenvolvido um outro processo de solução denominado Força Bruta, que consiste em percorrer com o trem-tipo toda estrutura por passos pré-estabelecidos e calcular os valores dos esforços limites. Observou-se que os resultados da EE podem ser melhorados quando se enriquece a solução com uma estratégia econômica de posicionamento de cargas concentradas em cima de picos da linha de influência. Para a grande maioria dos casos, os resultados obtidos com a EE foram corretos, porém, em alguns casos mais críticos, o valor exato da envoltória não é encontrado em algumas seções da estrutura, embora encontre um valor muito próximo a ele.

Palavras chave: Estratégia Evolutiva, Computação Evolucionária, Envoltória de Esforços Internos, Trem-tipo.

1 INTRODUÇÃO

A envoltória de esforços internos de uma estrutura é um diagrama que representa, para cada seção transversal da estrutura, os esforços limites devido à ação de um trem-tipo. Sendo assim, existe um diferente problema de otimização a ser resolvido em cada uma dessas seções.

Como não existe uma função matemática que descreva a variação de um determinado esforço em função da posição do trem-tipo, que no caso do problema em questão representaria a função objetivo, torna-se impossível o uso da maioria dos Métodos Clássicos de otimização, pois estes geralmente requerem o uso de derivadas da função objetivo. Portanto, para a solução deste problema, optou-se por utilizar um algoritmo de Estratégia Evolutiva (*EE*), que é um método probabilístico.

Para utilizar um algoritmo de *EE* é necessária uma função aptidão, que é responsável por indicar a qualidade das soluções durante o processo de otimização. Como neste caso o algoritmo de *EE* é utilizado em um problema de otimização, a função aptidão equivale à função objetivo.

Embora não exista uma função matemática que descreva a função aptidão, seu valor pode ser obtido com o auxílio de Linhas de Influência (*LI*), que descrevem a variação de um determinado efeito (por exemplo, uma reação de apoio, um esforço cortante ou um momento fletor em uma seção) em função da posição de uma carga vertical unitária que passeia sobre a estrutura [1], [2].

A avaliação da função aptidão é repetida inúmeras vezes durante o processo de otimização. Portanto, visando diminuir o esforço computacional envolvido nestes cálculos, ao se iniciar o processo de otimização em uma determinada seção transversal faz-se uma análise da *LI* e armazena-se as informações referentes a alguns de seus pontos de maior importância. Nestes pontos, denominados eventos (Figura 1), ocorrem:

- valores máximos ou mínimos;
- mudanças de sinal, isto é, pontos que a LI tenha valor nulo;
- descontinuidades;
- o início ou o fim de um trecho de valor constante;
- o início ou o fim da própria LI.

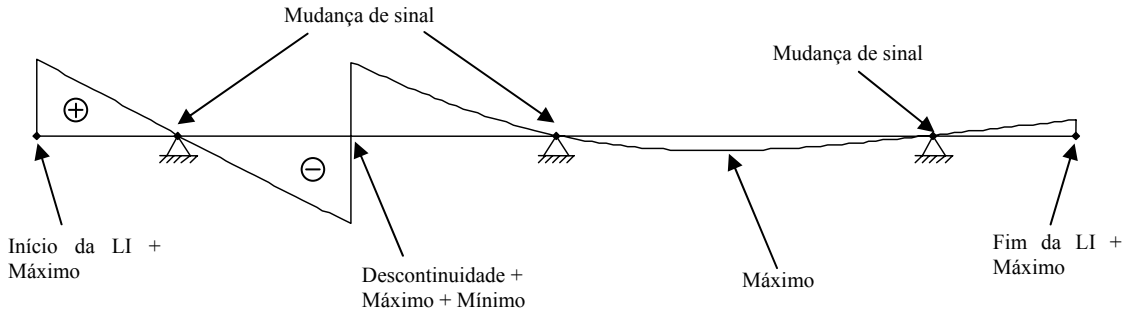

Figura 1 – Linha de influência com a identificação dos eventos.

2 ESTRATÉGIA EVOLUTIVA (*EE*)

Estratégia Evolutiva (*EE*) é um dos ramos da Computação Evolucionária (*CE*), que constitui uma classe de métodos de otimização probabilísticos inspirados por alguns princípios baseados em mecanismos evolutivos encontrados na natureza, como auto-organização e o comportamento adaptativo [3]. A principal idéia que está por trás de qualquer variação de um Algoritmo Evolucionário (*AE*) é: dada uma população de indivíduos, a pressão do meio ambiente causa uma seleção natural que evolui a população. Sendo assim, qualquer algoritmo evolucionário deve ter as seguintes componentes básicas para resolver um problema [4], [5], [6], [7]:

- **Uma representação genética das soluções do problema**

A representação ou codificação de um indivíduo quando se utiliza um *AE* consiste em relacionar o espaço real do problema com o espaço adotado pelo *AE*, ou seja, representar/codificar os elementos do espaço real no espaço do *AE*. Cada elemento do espaço de busca é denominado fenótipo e sua representação no espaço do *AE* é denominado genótipo. Na *EE*, cada indivíduo é representado por um par de vetores reais da forma $\boldsymbol{v} = (\boldsymbol{x}, \boldsymbol{\sigma})$, onde $\boldsymbol{x}$ representa um ponto de busca no espaço, ou seja, é o vetor das variáveis da função objetivo, e $\boldsymbol{\sigma}$ o vetor de desvio padrão associado.

- **População**

A definição de uma população equivale a decidir o número de indivíduos que irão formá-la. O papel da população é manter as possíveis soluções.

- **Uma maneira de inicializar a população**

A inicialização da população geralmente é simples na maioria das aplicações de *AE* e é feita gerando-se indivíduos aleatoriamente. Porém, algumas heurísticas podem ser usadas para gerar uma população inicial com maior aptidão, como, por exemplo, iniciar a população com soluções aproximadas conhecidas ou contendo algum tipo de informação prévia. Se isso vale o esforço computacional extra envolvido, depende muito da aplicação.

- **Uma função de aptidão**

A função aptidão é a responsável pelo processo de seleção dos indivíduos e deve indicar a qualidade de cada indivíduo na população, sendo assim, influi diretamente na evolução da população. Tecnicamente, é uma função que designa uma medida de qualidade ao genótipo, ou seja, a aptidão.

- **Operadores genéticos**

Os operadores genéticos alteram a composição genética dos filhos durante a reprodução. O papel dos operadores é criar novos indivíduos a partir dos antigos. Os principais operadores são recombinação e mutação.

A recombinação é um operador que une informações de dois ou mais genótipos pais para gerar um ou dois descendentes. Exemplos típicos de recombinação na *EE* é a recombinação discreta, onde cada componente do vetor dos descendentes é escolhido randomicamente das variáveis dos pais, e recombinação intermediária, onde cada componente é obtida através da média aritmética das variáveis dos pais.

A mutação é um operador que após ser aplicado a um genótipo gera um filho. Similar a recombinação, a mutação é um operador sempre estocástico: seu resultado – o filho – depende dos resultados de uma série de escolhas aleatórias.

Na *EE*, o operador de mutação pode ser aplicado ao vetor das variáveis ($\boldsymbol{x}$) e ao vetor do desvio padrão ($\boldsymbol{\sigma}$). A aplicação da mutação ao desvio padrão constitui o conceito de auto-adaptação [7], [8], que pode ser aplicado através das equações (1), (2) e (3).

$$\sigma_i' = \sigma_i . exp(\tau' . N(0,1) + \tau . N_i(0,1)) \tag{1}$$

$$\tau' = (\sqrt{2n})^{-1} \tag{2}$$

$$\tau = \left(\sqrt{2\sqrt{n}}\right)^{-1} \tag{3}$$

A mutação das variáveis ocorre geralmente a partir de uma perturbação gaussiana de média zero e um determinado desvio padrão, conforme equação (4).

$$x_i^{'} = x_i + N(0, \sigma_i^{'}) \tag{4}$$

onde:
i = 1,...,n ; onde n é o número de variáveis da função objetivo.
$N(.,.)$ é uma variável aleatória normal ou gaussiana.

- **Uma função de aptidão**

O papel da seleção é diferenciar os indivíduos baseados nas suas qualidades, em particular, permitir que os melhores indivíduos tornem-se pais da próxima geração. Atualmente, os dois principais tipos de *EE* são:

- $(\mu + \lambda) - EE$

Conhecida como estratégia soma, onde μ pais produzem λ filhos, sendo $\lambda > \mu$, gerando uma população de $\mu + \lambda$ indivíduos. Nesta estratégia, os $\mu + \lambda$ indivíduos participam do processo de seleção, que determina os μ indivíduos que serão os pais da próxima geração.

- $(\mu, \lambda) - EE$

Conhecida como estratégia vírgula, se difere da estratégia soma porque apenas os λ filhos participam do processo de seleção. Assim, o período de vida de cada indivíduo é limitado a apenas uma geração. Segundo [9], este tipo de estratégia tem bom desempenho em problemas onde o ponto ótimo é em função do tempo, ou onde a função é afetada por ruído.

- **Um critério de parada**

As opções comumente usadas como critério de parada são:

I. tempo máximo transcorrido;
II. o número total de avaliações da função aptidão atingir um valor limite;
III. quando a aptidão melhorar muito pouco durante um certo período de tempo (ou um certo número de gerações ou um certo número de avaliações da função aptidão);
IV. quando a diversidade da população diminuir até um certo limite, sendo diversidade uma medida do número de diferentes soluções presente na população, que pode ser medido pelas diferentes aptidões presentes na população ou pelo número de diferentes fenótipos ou genótipos presentes.

3 ALGORÍTMOS IMPLEMENTADOS

Como existem variações entre os algoritmos de *EE*, a implementação foi feita de duas maneiras distintas, de forma que fosse possível avaliar qual apresentaria melhor comportamento ao lidar com o problema de determinação de esforços críticos em função da posição do trem-tipo. Foi implementada a estratégia soma com apenas um indivíduo na

popupulação, que foi chamada de $(1+\lambda)-EE$, e com μ indivíduos, referenciada como $(\mu+\lambda)-EE$. Além de utilizar algoritmos de *EE* para resolver o problema de envoltórias, foi desenvolvido um outro método de solução denominado Força Bruta. Para refinamento dos resultados obtidos por esses métodos foi desenvolvido o processo Cargas-em-Picos, que é descrito na seqüência.

3.1 $(1+\lambda)-EE$

Trata-se de um algoritmo de *EE* onde há apenas um indivíduo na população que se reproduz e gera λ descendentes. Essa estratégia foi implementada utilizando heurísticas relacionadas a envoltórias de esforços.

Por se tratar de um problema unidimensional, foi possível fazer a busca dividindo-se o espaço de busca em trechos, que são intervalos da *LI* limitados por eventos. Para cada trecho determina-se os esforços máximos e mínimos locais e, no final da busca nos trechos, os esforços limites globais são determinados a partir dos resultados locais obtidos. Outra particularidade é que o posicionamento do trem-tipo foi feito a partir da posição da maior carga concentrada, sendo que quando houver mais de uma carga com o mesmo valor, a referência será a carga mais próxima à origem do trem-tipo.

A inicialização da população é feita com possíveis soluções do problema. Ao traçar manualmente a envoltória de esforços de uma determinada estrutura sujeita a ação de um trem-tipo que possua cargas concentradas, é intuitivo posicionar a maior carga concentrada sobre os eventos da *LI*, pois em grande parte dos casos, são as posições mais desfavoráveis à estrutura, isto é, posições que causam os maiores esforços. Então, como os trechos são limitados por eventos e como nesta estratégia a população possui apenas um indivíduo, a busca em cada trecho inicia-se com um indivíduo que represente a maior carga concentrada do trem-tipo posicionada no início do trecho.

Na literatura [8], [10], encontram-se recomendações para utilizar valores de λ/μ entre cinco e sete quando o valor de μ não for muito pequeno. Como nessa estratégia μ é igual a um, foi adotado λ igual a dez, que conforme testes realizados, foi o valor que apresentou melhores resultados.

O operador de mutação é aplicado ao indivíduo pai para gerar um novo indivíduo filho através da equação (4). Embora o operador de mutação não seja aplicado ao desvio padrão (σ), ele varia em função do comprimento de cada trecho, tendo sido estipulado um valor de 25% do comprimento, sendo limitado a um valor máximo, $\sigma_{max} = 4{,}00$, e a um mínimo, $\sigma_{min} = 0{,}20$, que fornecem uma precisão satisfatória no valor da posição crítica da carga móvel.

No processo de seleção, apenas o indivíduo que tiver a maior aptidão entre os $1+\lambda$ indivíduos presentes na população é selecionado para permanecer na próxima geração. Quando em uma determinada geração a população não se modificar, isto é, nenhum dos λ descendentes gerados apresentar melhor aptidão que o pai, é sinal que este ponto pode ser um máximo/mínimo local. Sendo assim, optou-se por fazer uma busca local, diminuindo o valor do desvio padrão para o valor σ_{min}, visando melhorar a qualidade da solução encontrada. Isso é mostrado na Figura 2, onde o primeiro ponto de máximo/mínimo local é indicado como 1º ponto limite. A busca prossegue com este valor de desvio padrão até a próxima geração que a população não se modificar, ou seja, até encontrar um próximo ponto limite. Feito isso, isola-se o espaço onde já foi realizada a busca dentro do trecho e reinicia-se a busca a partir de uma nova posição inicial, que é determinada a partir dos seguintes passos:

- após fazer a busca utilizando o valor σ calculado com base no comprimento do trecho e encontrar o primeiro ponto limite, marca-se uma suposta posição inicial para a próxima busca somando-se σ a posição deste ponto;
- continua-se a busca com o valor de σ_{min} e, após a determinação do novo ponto limite, calcula-se outra possível posição inicial somando-se σ_{min} ao novo ponto encontrado;
- a posição a partir da qual se reiniciará a busca será a posição mais a direita entre as duas posições calculadas anteriormente. A população será iniciada com um indivíduo correspondente a esta posição e voltará a ser usado o desvio padrão (σ).

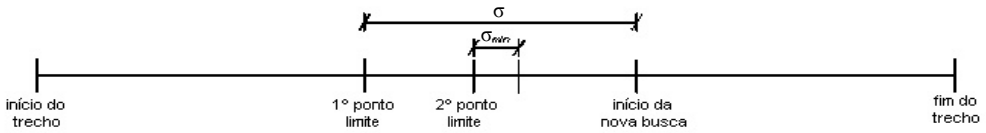

Figura 2 – Processo de busca por trechos.

3.2 $(\mu + \lambda) - EE$

Na estratégia $\mu + \lambda - EE$ não utilizou-se a sub-divisão do espaço de busca em trechos com a finalidade de avaliar o comportamento de um algoritmo de *EE* quando não lhe é fornecido informações adicionais do problema.

A inicialização da população foi feita de maneira aleatória, utilizando uma distribuição uniforme para gerar os indivíduos. Foi utilizado como ponto de referência para o posicionamento do trem-tipo a sua origem. O desvio padrão associado a cada indivíduo foi inicializado com o mesmo valor para todos os indivíduos da população. Este valor foi estipulado como sendo 10% do comprimento total do espaço de busca, que equivale a soma do comprimento do trem-tipo (l_t) com o comprimento da *LI* (l).

O único operador utilizado foi a mutação, que também foi aplicado ao desvio padrão, ou seja, empregou-se auto-adaptação do desvio padrão. Para isso, utilizou-se as equações (1), (2), (3) e (4).

Foram implementados os seguintes tipos de seleção:

- **Seleção global**

A seleção foi realizada fazendo uma concorrência entre todos os $\mu + \lambda$ indivíduos da população, de forma que os μ indivíduos que apresentassem o melhor aptidão fossem selecionados para serem os genitores da próxima população. Porém, observou-se que certas vezes esse processo de seleção causava uma diminuição prematura da diversidade da população, prejudicando o processo evolutivo. Uma das causas observadas é que quando o tamanho da população (μ) é pequeno, ela pode não representar todo o espaço de busca, levando a uma convergência para um máximo ou mínimo local.

- **Seleção individual**

Cada genitor compete apenas com seus descendentes para permanecer na população, através da avaliação da aptidão. É como se os indivíduos dessa população evoluíssem paralelamente.

Os dois tipos de seleção tiveram bons resultados quando os parâmetros foram adequadamente estabelecidos. Ou seja, na seleção global é mais importante a população ter

maior número de indivíduos (μ) do que continuar o processo evolutivo por um maior número de gerações (n_{ger}). Já na seleção individual, n_{ger} exerce papel fundamental. Com base nessas propriedades foram estabelecidos os valores para os parâmetros segundo a Tabela (1). O critério de parada adotado foi o número de gerações.

Tabela 2.1 – Parâmetros adotados na $(\mu + \lambda) - ES$.

Parâmetros	*Seleção*	
	Global	*Individual*
n_{ger}	15	40
μ	20	7
λ	60	21

3.3 Força Bruta

O processo que foi denominado Força Bruta não é um algoritmo evolucionário e nem propriamente um método de otimização. Consiste apenas em percorrer toda estrutura por passos de tamanho pré-estabelecido com o trem-tipo e calcular os valores dos esforços máximo e mínimo. Caso o trem-tipo não seja simétrico, esse processo é realizado percorrendo a estrutura com o trem-tipo nos dois sentidos. No final são determinados os esforços críticos na seção transversal em questão.

Neste processo, a precisão dos resultados depende do tamanho do passo (Δ), ou seja, quando menor o valor de Δ maior a chance de se obter bons resultados. Porém, Δ é um fator que influência o número de vezes que serão calculados os esforços, logo, quanto menor Δ, maior o número de vezes que o cálculo dos esforços será repetido e, conseqüentemente, mais caro fica o processo.

A determinação do Δ a ser usado foi a grande dificuldade encontrada para utilizar este método. A princípio pensou-se em estabelecer um valor a ser usado em todos os casos, porém, observou-se que o valor ótimo de Δ variava principalmente com o comprimento da *LI* (l).

Sendo assim, estabeleceu-se uma relação entre l e o valor de Δ a ser utilizado em cada problema, conforme a equação (5).

$$\Delta = 0{,}0025\, l \tag{5}$$

3.4 Cargas-em-Picos

O Carga-em-Picos não é um método de otimização. Consiste em calcular o valor do esforço considerando cada carga concentrada posicionada sobre os picos da *LI* e sobre o início e o fim de intervalos de valor constante, que podem ser identificados através dos eventos.

4 EXEMPLOS

4.1 Exemplo 1

Este exemplo mostra uma ponte rodoviária (Figura 3) submetida à ação do trem-tipo rodoviário da Figura 4, que é uma simplificação do trem-tipo bidimensional Classe 45 [11].

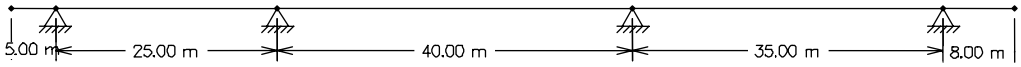

Figura 3 – Processo de busca por trechos.

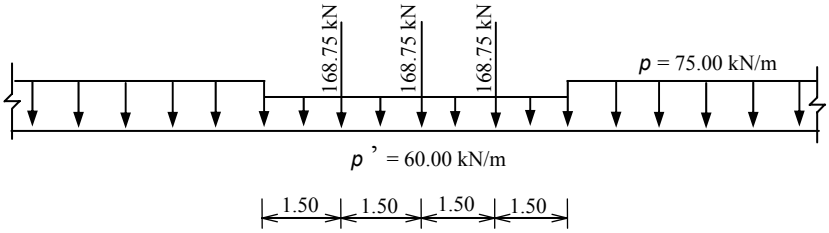

Figura 4 – Trem-tipo unidimensional resultante da transformação do trem-tipo bidimensional classe 45 [11].

Com os parâmetros adotados, todos os algoritmos implementados apresentaram ótimos resultados e, com isso, uma mesma configuração das envoltórias de esforço cortante e momento fletor, as quais são mostradas nas Figuras 5 e 6, respectivamente.

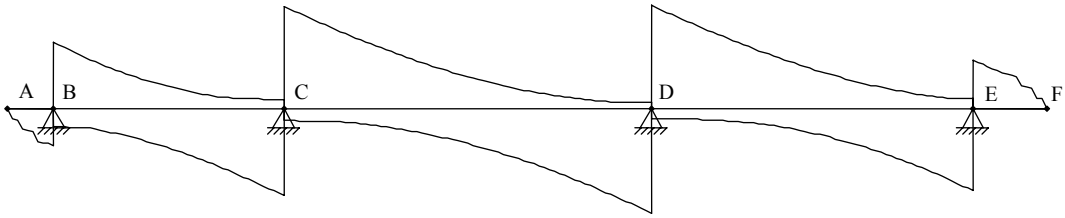

Figura 5 – Envoltória de esforço cortante do Exemplo 1 para $1+\lambda-EE$, $\mu+\lambda-EE$, Força Bruta e Cargas-em-Picos.

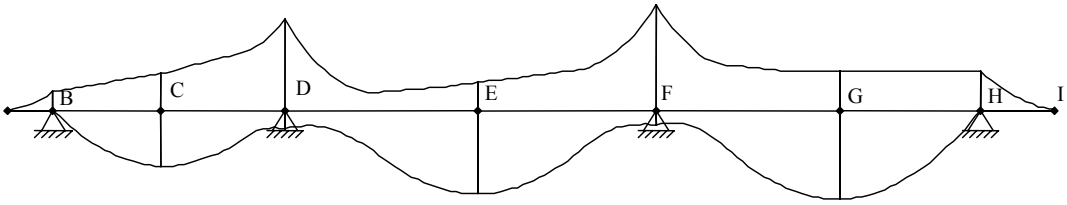

Figura 6 – Envoltória de momento fletor do Exemplo 1 para $1+\lambda-EE$, $\mu+\lambda-EE$, Força Bruta e Cargas-em-Picos.

Pode-se observar na Tabela 2 os valores da envoltória de esforço cortante nas seções pré-selecionadas e na Tabela 3 o erro em cada uma dessas seções. Embora a estratégia Força Bruta tenha apresentado os maiores erros, eles foram muito pequenos, sempre menores que 1% em todas as seções.

Tabela 2 – Resultados (kN) obtidos na envoltória de esforço cortante do Exemplo 1.

	$1+\lambda-EE$		$\mu+\lambda-EE$		Força Bruta		Cargas-em-Picos	
	max	min	max	min	max	min	max	min
B^{esq}	0.00	-813.90	0.00	-812.13	0.00	-813.86	0.00	-813.86
B^{dir}	1365.28	-368.00	1364.43	-367.98	1361.61	-367.98	1361.61	-367.98
C^{esq}	193.95	-1760.76	193.61	-1760.85	193.45	-1757.26	193.45	-1757.26
C^{dir}	2060.64	-243.43	2060.57	-243.39	2055.26	-243.39	2055.26	-243.39
D^{esq}	123.99	-2090.89	123.98	-2090.88	123.97	-2090.43	123.97	-2090.43
D^{dir}	2082.31	-208.75	2082.28	-208.10	2080.55	-207.00	2080.55	-207.00
E^{esq}	212.66	-1666.78	212.64	-1667.06	212.63	-1658.70	212.63	-1658.70
E^{dir}	1038.90	0.00	1038.86	0.00	1037.59	0.00	1037.59	0.00

Tabela 3 – Erros (%) obtidos na envoltória de esforço cortante do Exemplo 1.

	$1+\lambda-EE$		$\mu+\lambda-EE$		Força Bruta		Cargas-em-Picos	
	max	min	max	min	max	min	max	min
B^{esq}	-	-	-	-	-	-	-	-
B^{dir}	0.001	0.000	0.063	0.005	0.270	0.005	0.000	0.005
C^{esq}	0.000	0.012	0.175	0.007	0.258	0.211	0.005	0.000
C^{dir}	0.000	0.000	0.003	0.016	0.261	0.016	0.000	0.016
D^{esq}	0.000	0.008	0.008	0.009	0.016	0.030	0.008	0.000
D^{dir}	0.000	0.000	0.001	0.311	0.085	0.838	0.001	0.000
E^{esq}	0.000	0.017	0.009	0.000	0.014	0.501	0.014	0.000
E^{dir}	0.000	-	0.004	-	0.126	-	0.000	-

Os resultados obtidos nas seções pré-selecionadas da envoltória de momento fletor também foram satisfatórios, apresentando erros muito pequenos. Similar a envoltória de esforços cortantes, os momentos fletores limites ocorreram sempre ao se posicionar alguma carga concentrada sobre os picos ou limites de intervalos de valor constante da *LI*. Isso pode ser concluído ao observar que o erro na estratégia Cargas-em-Picos é praticamente nulo.

Ao se utilizar a estratégia $(\mu+\lambda)-EE$, algumas vezes surgiram falhas ao longo da envoltória de momento fletor, como a da Figura 7. Essas falhas representam seções da estrutura onde o número de gerações estabelecido como critério de parada não foi suficiente para que o algoritmo pudesse encontrar o esforço crítico. Por ser a falha visivelmente detectada e por ela não se repetir ao solicitar novamente o cálculo da envoltória, esse tipo de falha não desqualifica esse método para a determinação de envoltórias.

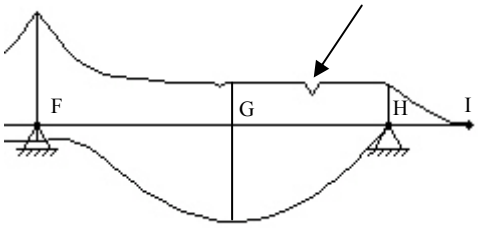

Figura 7 – Falha na envoltória de momento fletor ao utilizar a estratégia $\mu+\lambda$.

4.2 Exemplo 2

Para avaliar o comportamento dos algoritmos implementados ao lidar com estruturas com dimensões relativamente pequenas, foi feito um exemplo com a estrutura da Figura 8 e o trem-tipo rodoviário da Figura 4.

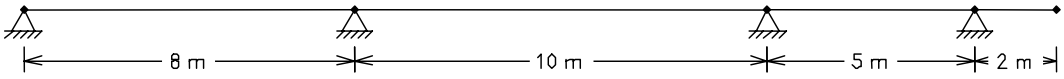

Figura 8 – Estrutura do Exemplo 2.

Como conseqüência do bom funcionamento com estruturas de dimensões menores que as usuais, a configuração obtida para as envoltórias de esforço cortante e momento fletor, Figura 9 e Figura 10 respectivamente, foi a mesma ao utilizar os algoritmos $1+\lambda-EE$, $\mu+\lambda-EE$, Força Bruta e Cargas-em Picos.

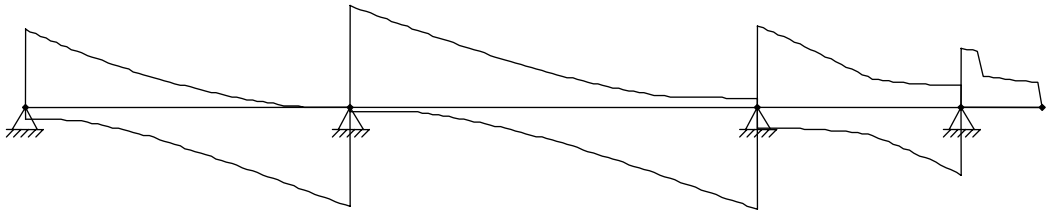

Figura 9 – Envoltória de esforço cortante do Exemplo 2 para $1+\lambda-EE$, $\mu+\lambda-EE$, Força Bruta e Cargas-em-Picos.

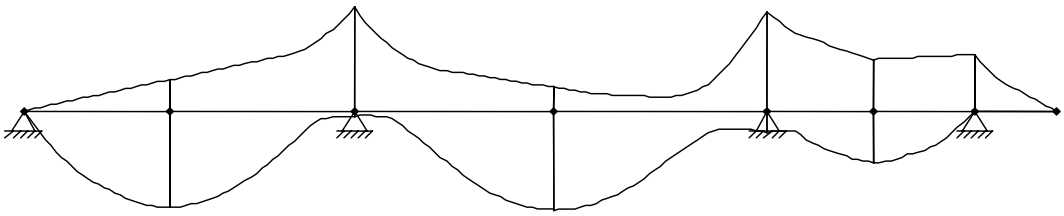

Figura 10 – Envoltória de momento fletor do Exemplo 2 para $1+\lambda-EE$, $\mu+\lambda-EE$, Força Bruta e Cargas-em-Picos.

Uma situação particular ocorre no balanço dessa estrutura ao traçar a envoltória de esforço cortante utilizando o trem-tipo da Figura 11. O resultado obtido quando foi utilizado apenas a estratégia Cargas-em-Picos está na Figura 12a. Para os demais métodos, a Figura 12b mostra o resultado.

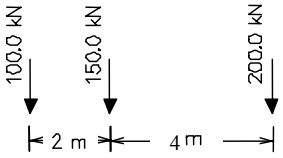

Figura 11 – Trem-tipo.

Figura 12 – Envoltória de esforço cortante no balanço da estrutura do Exemplo 2 utilizando o trem-tipo da Figura 11.

Apenas a estratégia Cargas-em-Picos é capaz de sempre detectar quando há esse tipo de pico na envoltória. Nos métodos evolutivos a posição crítica do trem-tipo é encontrada porque durante o processo são geradas posições próximas à critica, as quais geralmente possuem uma alta aptidão, que ajudam na evolução da população. Neste caso isso não ocorre, pois nenhuma posição, por mais próxima que seja da posição crítica, causa um valor da envoltória próximo ao que ocorre na posição crítica. Sendo assim, é muito pequena a probabilidade de gerar aleatoriamente indivíduos que correspondam exatamente a esta posição crítica.

Na estratégia Força Bruta ocorre o mesmo problema, pois também é muito pequena a probabilidade de se posicionar o trem-tipo exatamente na posição crítica a partir do Δ utilizado.

5 ANÁLISE DO CUSTO COMPUTACIONAL

Para analisar a variação do número de avaliações da função aptidão, a Figura 13 mostra um gráfico com o número de avaliações da função aptidão (n_{tot}) realizadas para traçar a envoltória de esforço cortante máximo nos exemplos anteriores e a Figura 14 mostra a variação tempo de processamento gasto no cálculo da envoltória de esforços cortantes, incluindo o tempo necessário para o cálculo da *LI*. Foi utilizado um computador com processador Pentium III de 1,1 GHz e 512 Mb de memória RAM.

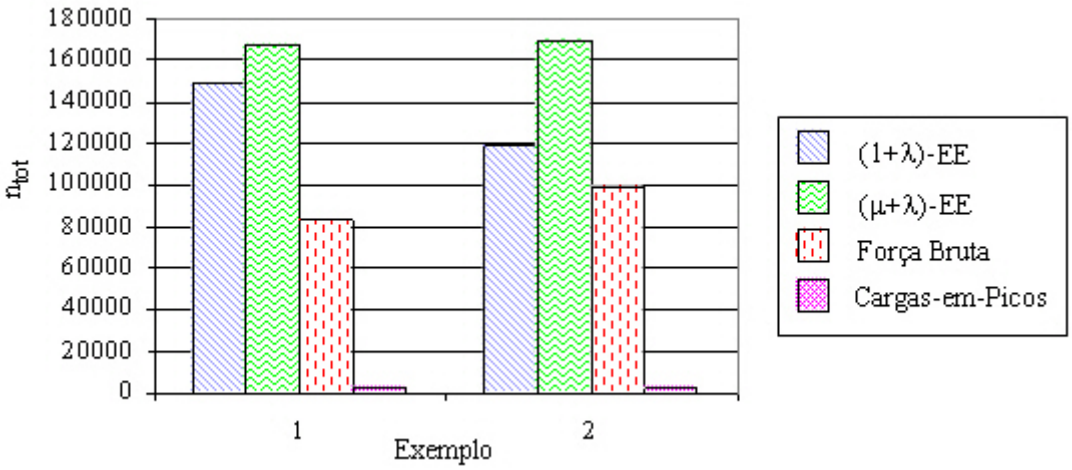

Figura 13 – Número de avaliações da função aptidão na envoltória de esforço cortante máximo.

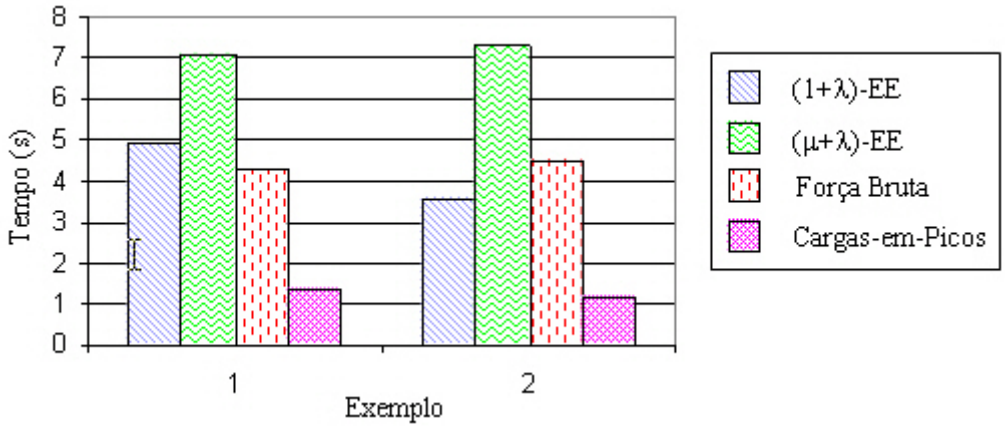

Figura 14 – Tempo de processamento do programa para cálculo da envoltória de esforço cortante máximo.

6 CONCLUSÕES

Nos exemplos analisados, tendo a estrutura pequenas ou grandes dimensões, todos os algoritmos tiveram ótimos resultados. Por ser o método que não utilizou informações adicionais relacionadas ao problema de envoltória de esforços, o método de $\mu + \lambda - EE$ foi o que apresentou maiores variações nos resultados, embora essas variações sejam muito pequenas.

No Força Bruta não aparecem falhas visíveis, embora muitas vezes não se consiga determinar os resultados corretos. Sua desvantagem é que toda vez que se faz a determinação da envoltória de um esforço usando um certo Δ, sempre se encontra o mesmo resultado. Sendo assim, mesmo quando o resultado obtido não seja o correto, ele se repetirá toda vez que calcular a envoltória. Embora os métodos $1 + \lambda - EE$ e $\mu + \lambda - EE$ estejam sujeitos ao surgimento de falhas visíveis, elas raramente repetem-se numa nova solicitação de cálculo da envoltória.

O Cargas-em-Picos justificou seu uso para refinamento dos resultados, pois na maioria dos casos o esforço limite ocorre com alguma carga concentrada posicionada sobre os picos ou limites de intervalos de valor constante da *LI* e, em certas situações, é o único método capaz de determinar o valor exato do esforço. Além disso, apresentou um custo computacional muito baixo, pois seu tempo de processamento é praticamente desprezível em relação aos outros métodos.

A partir dessas observações, observou-se que o uso do método $1 + \lambda - EE$ em conjunto com o Cargas-em-Picos seria o melhor método a ser usado.

A maior dificuldade encontrada em todos os métodos foi estimar os parâmetros a serem utilizados, tais como desvio padrão (σ), tamanho da população (μ), número de descendentes (λ) a serem gerados e o valor de Δ. Para a grande maioria dos casos, os resultados obtidos foram exatos, porém, em alguns casos mais críticos, o valor exato da envoltória não é encontrado em algumas seções da estrutura, embora se encontre um valor muito próximo a ele.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] MARTHA, L. F.– **Métodos Básicos da Análise de Estruturas.** Disponível em: < http://www.tecgraf.puc-rio.br/~lfm >. Acesso em: 03 abr. 2005.

[2] SÜSSEKIND, J. C. **Curso de Análise Estrutural**: Estruturas Isostáticas – v. 2, Editora Globo, Porto Alegre, 1997.

[3] BEYER, H.-G.; SCHWEFEL, H.-P; WEGENER, I. **How to Analyse Evolutionary Algorithms.** Theoretical Computer Science, v.287, p. 101-130, set. 2002.

[4] MICHALEWICZ, Z. **Evolutionary Computation: Practical Issues.** Proceedings of 1996 IEEE International Conference on Evolutionary Computation, IEEE Press, Florida, p. 57-62, 1996.

[5] EIBEN, A. E.; SMITH, J. E. **Introduction to Evolutionary Computing**. Springer, Cap. 2, p. 15-35, 2003. ISBN: 3-540-40184-9. Disponível em: < http://www.cs.vu.nl/%7Egusz/ecbook/Eiben-Smith-Intro2EC-Ch2.pdf >. Acesso em: 03 abr. 2005.

[6] BARBOSA, H. J. C., **Algoritmos Genéticos para Otimização em Engenharia: Uma Introdução aos Algoritmos Genéticos.** 2a Escola de Verão em Computação Científica, LNCC, Rio de Janeiro, RJ, 1997.

[7] BÄCK, T.; HAMMEL, U.; SCHWEFEL, H.-P. **Evolutionary Computation: Comments on the History and Current State.** IEEE Transaction on Evolutionary Computation, v. 1, n. 2, abr. 1997.

[8] BÄCK, T.; HAMMEL, U. **Evolution Strategies Applied to Perturbed Objective Functions.** Congress on Evolutionary Computation, p. 40-45, 1994.

[9] CORTES, O. A. C.; SAAVEDRA, O. R. **Estratégias Evolutivas Paralelas em Otimização Multimodal.** INFOCOMP – Revista de Ciência da Computação – Anais da III SECICOM, v. 2, n. 1, p. 63-68, nov. 2000.

[10] COSTA, L.; OLIVEIRA, P. **An Evolution Strategy for Multiobjective Optimization.** Proceedings of the CEC 2002 - Congress on Evolutionary Computation, Honolulu, USA, 2002.

[11] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT) – NBR – 7188 – **Carga móvel em ponte rodoviária e passarela de pedestre.** Rio de Janeiro, 1982.