

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
DEPARTAMENTO DE ENGENHARIA MECÂNICA

OTIMIZAÇÃO DE MASSA DE TRELIÇAS UTILIZANDO O ALGORITMO HARMONY SEARCH

por

Matheus Lopes da Silva

Monografia apresentada ao Departamento de Engenharia Mecânica da Escola de Engenharia da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para obtenção do diploma de Engenheiro Mecânico.

Porto Alegre, junho de 2011.



Universidade Federal do Rio Grande do Sul
Escola de Engenharia
Departamento de Engenharia Mecânica

OTIMIZAÇÃO DE MASSA DE TRELIÇAS UTILIZANDO O ALGORITMO HARMONY SEARCH

por

Matheus Lopes da Silva

ESTA MONOGRAFIA FOI JULGADA ADEQUADA COMO PARTE DOS
REQUISITOS PARA A OBTENÇÃO DO TÍTULO DE
ENGENHEIRO MECÂNICO
APROVADA EM SUA FORMA FINAL PELA BANCA EXAMINADORA DO
DEPARTAMENTO DE ENGENHARIA MECÂNICA

Prof. Arnaldo Ruben Gonzalez
Coordenador do Curso de Engenharia Mecânica

Área de Concentração: **Mecânica dos Sólidos**

Orientador: Prof^a. Dr^a. Letícia Fleck Fadel Miguel

Comissão de Avaliação:

Prof. Dr. Walter Jesus Paucar Casas

Prof. Dr. Jun Sérgio Ono Fonseca

Prof. Dr. Rogério José Marczak

Porto Alegre, 27 de junho de 2011.

Dedico esse trabalho a meus pais, Derli e Zile, às minhas irmãs, meu irmão e minha noiva. Agradeço pelo apoio, incentivo e compreensão durante esses anos de estudo. Sou grato por todos os ensinamentos, exemplos e oportunidades que tive em minha vida graças à presença e ajuda incondicional de vocês.

AGRADECIMENTOS

Em primeiro lugar agradeço à minha família, em especial aos meus pais, Derli e Zile, às minhas irmãs, ao meu irmão, à minha noiva.

À minha orientadora, Letícia Fleck Fadel Miguel, pela ajuda e compreensão durante o período de desenvolvimento deste trabalho.

Aos professores da Universidade Federal do Rio Grande do Sul que sabem a importância de formar profissionais bem qualificados e que tratam o compromisso de ensinar e o desenvolvimento científico e tecnológico como prioridade em suas vidas.

Aos meus colegas de aula, pelos anos de estudo e dedicação para nos tornarmos profissionais bem capacitados. Agradeço, também, pelos momentos de lazer e companheirismo que passamos juntos.

Aos meus colegas de trabalho, pela experiência e conhecimento passados e também por momentos de descontração e alegria. Agradeço às empresas que trabalhei por acreditarem no meu potencial e, através de desafios, possibilitaram que eu crescesse não só profissionalmente, mas também como pessoa.

À sociedade brasileira por custear os meus estudos durante estes anos de faculdade e assim proporcionar o meu crescimento profissional e pessoal.

SILVA, M. L. **OTIMIZAÇÃO DE MASSA DE TRELIÇAS UTILIZANDO O ALGORITMO HARMONY SEARCH**. 2011. 17. Monografia (Trabalho de Conclusão do Curso em Engenharia Mecânica) – Departamento de Engenharia Mecânica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2011.

RESUMO

Na sociedade atual está sendo cada vez mais valorizada a otimização de recursos, essa prática não está restrita a alguma área do conhecimento. Há necessidade de se fazer mais e melhor utilizando menos recursos e insumos com igual ou maior qualidade e segurança. A área da engenharia utiliza os mais variados métodos para otimizar massa, distribuição de material, custos, etc. Neste contexto, no presente trabalho utiliza-se do método meta-heurístico denominado *Harmony Search* para otimizar a massa de estruturas com restrições de tensões e deslocamentos. Foi utilizado um *software* para a determinação dos esforços internos da estrutura e outro para a implementação do algoritmo propriamente dito. Para comparar os resultados obtidos neste trabalho com casos da literatura utilizaram-se estruturas já bastante estudadas e que servem de referência para tornar possível o confronto entre soluções obtidas por métodos diferentes. Concluiu-se que o método estudado apresentou respostas semelhantes aos métodos da literatura para os problemas analisados, confirmando a eficácia do método, porém apresentou valores de massa ligeiramente superiores a alguns outros métodos.

PALAVRAS-CHAVE: *Harmony Search*, Otimização, Estruturas Metálicas.

SILVA, M. L. **OPTIMIZATION OF MASS OF TRUSS USING HARMONY SEARCH ALGORITHM**. 2011. 17. Monografia (Trabalho de Conclusão do Curso em Engenharia Mecânica) – Departamento de Engenharia Mecânica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2011.

ABSTRACT

In today's society it is being increasingly valued the resource optimization, this practice is not restricted to any area of knowledge. There is the need to do more, and better, using less resources and inputs, with equal or better quality and safety. The engineering field uses the most varied methods to optimize mass, distribution of materials, costs, and so on. In this context, this study makes use of the meta-heuristic method, called *Harmony Search*, to optimize the mass of the structures with restrictions in stress and in displacement. A software was used to determine the internal forces of the structure and another one for the implementation of the algorithm itself. To compare the results obtained in this study with cases from the literature, it was used structures which have already been much studied and that serve as reference to make possible the comparison between solutions obtained by different methods. It was concluded that the method studied showed similar responses to the literature methods for the analyzed problems, confirming the effectiveness of the method, however, it showed slightly higher mass values than in some other methods.

KEYWORDS: Harmony Search, Optimization, Metal Structures.

ÍNDICE

	Pág.
1. INTRODUÇÃO.....	1
2. OBJETIVOS.....	1
3. REVISÃO BIBLIOGRÁFICA.....	1
4. FUNDAMENTAÇÃO TEÓRICA.....	2
4.1. Harmony Search.....	2
5. METODOLOGIA.....	4
6. APRESENTAÇÃO DOS PROBLEMAS.....	4
6.1. Treliça bidimensional de 10 barras.....	4
6.2. Treliça bidimensional de 10 barras com restrições de deslocamento.....	6
6.3. Treliça tridimensional de 25 barras.....	7
6.4. Treliça tridimensional de 72 barras.....	10
7. CONCLUSÕES.....	12
REFERÊNCIAS BIBLIOGRÁFICAS.....	14
Anexo I.....	15

1. INTRODUÇÃO

Na atual sociedade em que se vive a palavra “otimização” vem sendo a cada dia mais valorizada. Podem-se citar várias áreas do conhecimento onde se precisa fazer mais com menos recursos, ou seja, naturalmente se buscam soluções mais eficientes para questões que já foram solucionadas de formas menos eficientes. Nos dias de hoje a otimização ganhou grande importância nas engenharias para a minimização de materiais, maximização de confiabilidade, evolução no projeto de estruturas, etc. Na engenharia podem-se citar exemplos em todas as suas áreas de atuação, e no ramo da Mecânica dos Sólidos não é diferente. Devem-se resolver problemas estruturais com a mesma ou maior confiabilidade aplicando de forma mais racional os recursos.

As estratégias de otimização baseadas em métodos meta-heurísticos têm sido massivamente utilizadas na solução de problemas nos mais variados campos do conhecimento. Podem ser citados como exemplo: Algoritmos Genéticos (Holland, 1975 e Goldberg, 1989), *Particle Swarm Optimization* (Kennedy e Eberhart, 1995), *Bees Algorithm* (Pham *et al.*, 2006), *Firefly Algorithm* (Xin-She Yang, 2008), entre vários outros.

Neste contexto, no presente trabalho é empregado o método de otimização denominado *Harmony Search*, desenvolvido por Geem *et al.* (2001) que desde então vem sendo estudado e aplicado em diversos problemas tais como minimização ou maximização de funções, projeto de treliças, redes de distribuição de água, entre outros (Xin-She Yang, 2008), para solucionar problemas de engenharia já bastante estudados e massivamente publicados na literatura técnica a fim de comparar o referido método com outros apresentados na literatura.

2. OBJETIVOS

O presente trabalho tem como objetivo entender, utilizar e validar o método meta-heurístico de otimização denominado *Harmony Search*.

O método será empregado para a otimização de massa de treliças “de referência” de duas e três dimensões, submetidas a restrições de tensões e deslocamentos, as quais já foram previamente otimizadas por outros pesquisadores através de outros métodos e por isso serão utilizadas para comparar e validar o *Harmony Search*. Portanto, a função objetivo é a massa mínima da treliça em questão e as restrições são os deslocamentos e tensões máximos admissíveis.

3. REVISÃO BIBLIOGRÁFICA

Durante a segunda guerra mundial começou-se a utilizar algoritmos heurísticos para decifrar códigos, e desde então o desenvolvimento da utilização dos algoritmos heurísticos para otimização tem crescido cada vez mais. As décadas de 1960 e 1970 foram muito importantes para a evolução dos métodos de otimização, na Universidade de Michigan, John Holland e seus colaboradores os quais desenvolveram os algoritmos genéticos. Em 1975 os algoritmos genéticos já tinham sido mais bem estudados e se tinha utilizado métodos de recombinação, cruzamentos e manipulações e assim foi publicado o livro de John Holland (1975) resumindo o desenvolvimento dos algoritmos genéticos.

Após o desenvolvimento dos algoritmos genéticos foram desenvolvidos inúmeros métodos de otimização, sendo que um dos mais atuais e promissores é o *Harmony Search*, o qual será utilizado no presente trabalho para otimização de treliças.

A otimização de funções está sendo cada vez mais estudada e difundida nas áreas do conhecimento, isso se deve à evolução natural do conhecimento e do próprio ser humano que está sendo levado a resolver problemas buscando a maior eficiência possível com o menor uso de recursos. Pode-se considerar que em todas as áreas há a necessidade de aprimorar soluções, de encontrar mínimos, como exemplo observa-se a área da administração onde se devem minimizar custos sem comprometer o negócio. Na engenharia é mais fácil dar exemplos nos quais se devem otimizar funções com as mais distintas variáveis.

Na área do conhecimento relativa à Engenharia Mecânica a busca do “ótimo” a cada dia vem sendo mais valorizada e por isso desenvolvida de forma muito expressiva. Na sua maioria, os problemas de engenharia mecânica podem ter suas soluções otimizadas e assim ter um uso mais racional dos recursos ligados aos referidos problemas.

Uma questão comum em projetos de estruturas é a minimização de massa, mas ainda atendendo limitações de normas e dos próprios projetos quanto a tensões, deslocamentos, vibrações, etc. Na literatura pode-se enumerar muitos métodos de otimização que podem ser utilizados com o intuito de minimizar a massa de estruturas metálicas mantendo a confiabilidade descrita em projeto. Segundo (Souza, 2010) pode-se classificar os métodos de otimização em determinísticos e probabilísticos, os métodos classificados como determinísticos tem como característica utilizar a primeira derivada da função em estudo, no entanto nos métodos probabilísticos não é necessário o uso da derivada da função e por este motivo também podem ser chamados de métodos de ordem zero.

Neste trabalho optou-se por trabalhar com um método probabilístico, os quais possuem como características:

- Possibilidade de paralelização dos algoritmos, pois trabalham com uma população de soluções;
- Não necessitam que a função objetivo seja diferenciável;
- Exige normalmente um maior custo computacional em relação aos determinísticos.

4. FUNDAMENTAÇÃO TEÓRICA

4.1. *Harmony Search*

Os métodos de otimização na sua grande maioria foram desenvolvidos através da observação de fenômenos presentes na natureza, nas sociedades, nas relações interespecíficas, etc, não sendo diferente no método estudado no presente trabalho, que é baseado na forma como músicos improvisam suas melodias. Um músico quando necessita improvisar uma melodia utiliza sua memória musical para executar uma nova combinação de notas e assim forma uma nova melodia com uma maior possibilidade de ser apreciada por seus espectadores.

Segundo Scalabrin *et al.* (2010), o *Harmony Search* pode ser dividido no seguintes passos:

I - Inicialização do Problema e Parâmetros do Algoritmo:

No primeiro passo, como em todo problema de otimização, é definida uma função objetivo a ser otimizada, a qual pode ou não possuir um conjunto de restrições. Originalmente o *Harmony Search* foi desenvolvido para a resolução de problemas de minimização (Geem *et al.*, 2001). Neste passo também são definidos os parâmetros do algoritmo. Os quatro principais parâmetros são o tamanho da memória harmônica (*Harmony Memory Size – HMS*), a taxa de escolha de um valor da memória (*Harmony Memory Considering Rate – HMCR*), a taxa de ajustes dos valores (*Pitch Adjusting Rate – PAR*) e número máximo de improvisos (*Maximum Improvisation – MI*).

II - Inicialização da Memória Harmônica:

No segundo passo é inicializada a Memória Harmônica (HM) com um número de harmonias geradas aleatoriamente. A Memória Harmônica é o vetor no qual são armazenadas as melhores harmonias encontradas durante uma determinada execução. Cada harmonia é um vetor que representa uma possível solução para o problema.

III - Improviso de uma nova Harmonia:

No terceiro passo é improvisada uma nova harmonia baseada nas harmonias existentes na HM, sendo que a nova harmonia é uma combinação de várias outras harmonias. Para cada

variável da nova harmonia seleciona-se arbitrariamente uma harmonia da HM, verificando a probabilidade deste valor ser ou não utilizado (HMCR). Se for utilizado um valor de outra harmonia, o valor desta variável pode sofrer pequenos ajustes (*Fret Width* – FW) segundo uma probabilidade (PAR). Se não for utilizado o valor de outra harmonia, um valor aleatório dentro do intervalo de valores permitidos é atribuído. Sendo assim, os parâmetros HMCR e PAR do algoritmo HS são responsáveis por estabelecer um balanço entre a busca global e a busca local no espaço de busca.

IV - Atualização da Memória Harmônica:

No quarto passo, a cada nova harmonia improvisada é verificada se ela é melhor do que a pior harmonia da HM. Caso confirmada esta condição, a nova harmonia substitui a pior harmonia da HM.

V - Verificação do critério de parada:

No quinto passo, ao término de cada iteração é verificado se a melhor harmonia satisfaz o critério de parada, normalmente número máximo de improvisos MI. Em caso positivo, a execução é terminada. Caso contrário, retorna para o segundo passo enquanto não atingir o critério de parada.

Parâmetros: *HMS, HMCR, PAR, MI, FW*

Início

Funcao Objetivo $f(\vec{x})$, $\mathbf{x} = [x_1, x_2, \dots, x_d]^T$

Inicializa a memoria harmonica \vec{x}_i $i = 1, 2, \dots, HMS$

Avalia cada harmonia na memoria harmonica : $f(\vec{x}_i)$

Ciclo = 1

Enquanto *Ciclo* < *MI*

Para *cada variavel* $j = 1, 2, \dots, N$

Se *aleatorio* $\leq HMCR$ {*Consideracao da Memoria*}

$x'_j = x_j^i$, *com* $i \in [1, HMS]$ *escolhido aleatoriamente*

Se *aleatorio* $\leq PAR$ {*Consideracao de Ajuste*}

$x'_j = x'_j \pm r \times FW$, *com* r *aleatorio*

fim se

Senão *Selecao Aleatoria*

Gera x'_j *aleatoriamente*

fim Para

Avalia a nova harmonia gerada : $f(\vec{x}')$

Se $f(\vec{x}')$ *eh melhor que a pior harmonia na HMS*

Atualiza a memoria harmonica

Ciclo = *Ciclo* + 1

fim Enquanto

Resultados e visualizacoes

fim

Figura 4.1 - Pseudo-código do *Harmony Search* (Scalabrin *et al.*, 2010).

5. METODOLOGIA

Para executar os testes e comparações do método de otimização estudado, serão usadas como base estruturas que se tornaram referência na literatura por terem sido estudadas e otimizadas aplicando diversos métodos de solução e dessa forma torna-se possível executar a comparação das soluções apresentadas neste trabalho. Serão estudados quatro casos, sendo que dois são treliças de três dimensões e dois casos de treliças de duas dimensões, e em todos os casos são impostas restrições de tensões e ou de deslocamentos.

Com o intuito de calcular os esforços internos das estruturas estudadas, foram utilizados dois *softwares* comerciais FrameTOOL e MASTAN sendo o primeiro para a estrutura de duas dimensões e o segundo para as demais. Já para a implementação do *Harmony Search* usou-se o MATLAB. O Teorema de Castigliano foi utilizado para a implementação das restrições de deslocamentos, sendo descrito, por exemplo, em Hibbeler (2004).

Por tratar-se de um método probabilístico, no qual o começo da resolução do problema se dá pela atribuição de valores randômicos para as possíveis soluções, foram realizadas 100 simulações para cada caso.

6. APRESENTAÇÃO DOS PROBLEMAS

6.1. Treliça bidimensional de 10 barras

A treliça bidimensional apresentada na Figura 6.1 é um dos casos de referência analisados com o intuito de comparar este estudo com outros já desenvolvidos anteriormente na literatura.

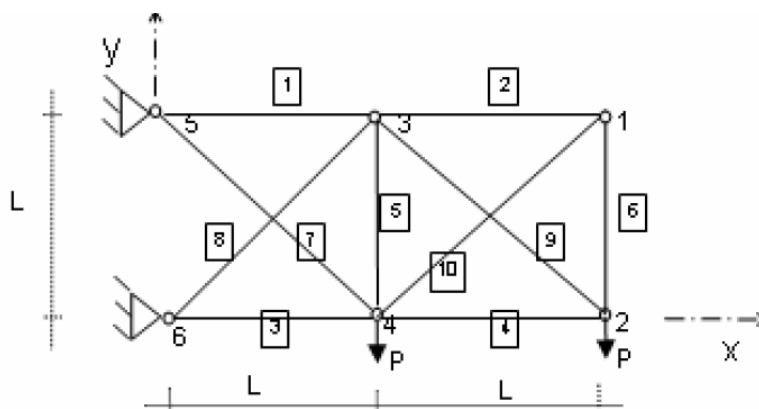


Figura 6.1 – Treliça de dez barras (Guerra, 2008).

As condições de contorno do problema bem como as propriedades do material da estrutura e suas características foram definidas de acordo com a literatura a fim de comparar a solução obtida neste trabalho com outras soluções já publicadas. A Tabela 6.1 apresenta os dados da treliça de 10 barras.

Tabela 6.1 – Dados do problema da treliça de 10 barras.

Módulo de elasticidade	$E = 68,9 \times 10^9 \text{ N/m}^2$
Massa específica	$\rho = 2768 \text{ kg/m}^3$
Comprimento dos módulos da estrutura	$L = 9,144 \text{ m}$
Carga	$P = 444,8 \text{ kN}$
Tensão admissível de tração e de compressão	$ \delta < 172 \times 10^6 \text{ N/m}^2$, exceto barra 9 onde $ \delta < 517 \times 10^6 \text{ N/m}^2$
Limite das variáveis de projeto	$6,451 \times 10^{-5} \text{ m}^2 < A_i < 6,451 \times 10^{-3} \text{ m}^2$

Onde A_i são os valores das áreas para cada barra da estrutura e assim sendo otimizada a área garante-se a otimização da massa da estrutura já que o comprimento da mesma não varia consideravelmente.

O primeiro passo para a solução do problema é obter os valores dos esforços internos e assim montar as restrições para o algoritmo de otimização, na Tabela 6.2 estão listadas as coordenadas dos nós da estrutura.

Tabela 6.2 – Coordenadas dos nós da estrutura.

Nó	Coordenada X(m)	Coordenada Y(m)
1	18,28	9,14
2	18,28	0
3	9,14	9,14
4	9,14	0
5	0	9,14
6	0	0

Foi utilizado o FrameTOOL, *software* para análise de estruturas, para obter os esforços internos e aplicar o método *Harmony Search* que foi implementado em linguagem MATLAB, assim foram otimizadas as áreas das barras que compõe a estrutura em questão.

Os parâmetros aplicados para a configuração do método foram os identificados como ótimos após algumas análises do método. Na Figura 6.2 pode-se ver o comportamento do algoritmo em relação ao número de iterações ao qual foi submetido.

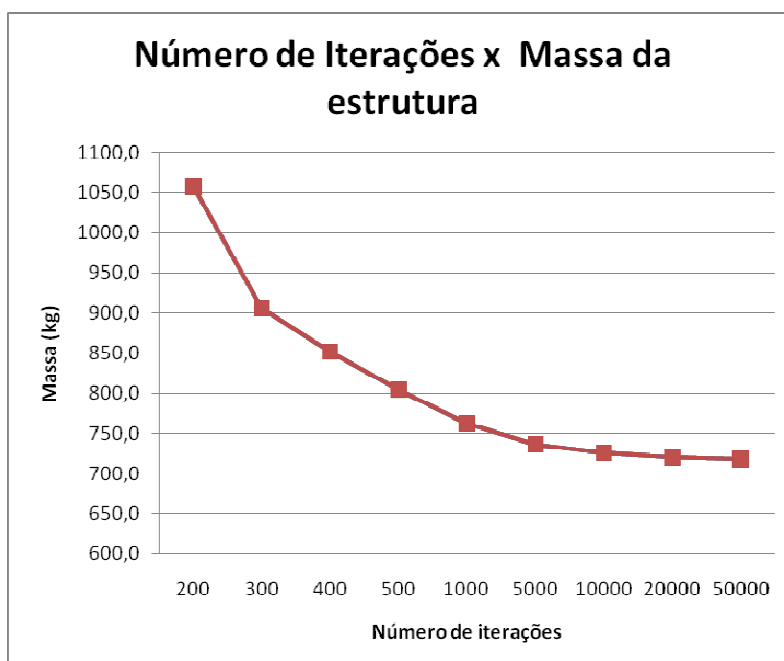


Figura 6.2 – Relação entre número de iterações e massa obtida.

Pode-se observar que o perfil assintótico, que já era esperado, se confirma quando se tem um aumento do número de iterações culminando para um valor de 717kg quando 50000 iterações são executadas.

A Tabela 6.3 mostra a comparação dos resultados obtidos utilizando o método apresentado no presente trabalho com soluções publicadas na literatura utilizando outros métodos.

Tabela 6.3 – Comparação entre as soluções.

Variáveis	SQP (<i>Sequential Quadratic Programming</i>)	G.A.s (<i>Ghasemi et al.,1999</i>)	<i>Harmony Search</i> (presente trabalho)
A1(mm ²)	5096,7	4612,8	5045,47
A2(mm ²)	64,5	295,4	1033,27
A3(mm ²)	64,5	5443,8	5294,95
A4(mm ²)	2526,1	2286,4	1556,01
A5(mm ²)	64,5	64,5	924,15
A6(mm ²)	64,5	296,7	1042,08
A7(mm ²)	3741,9	4056,1	3819,13
A8(mm ²)	3554,8	3220,6	3483,28
A9(mm ²)	2374,1	2161,2	738,91
A10(mm ²)	90,32	416,1	1461,98
Massa (kg)	679,03	687,65	717,60

Observando o resultado e comparando com as soluções apresentadas na literatura pode-se chegar à conclusão que para esta estrutura o método escolhido não foi o melhor, pois a massa total da estrutura resultou aproximadamente 5% maior que a massa obtida com os dois métodos apresentados na literatura. O algoritmo utilizado neste trabalho para a otimização deste problema está descrito no Anexo I.

6.2. Treliça bidimensional de 10 barras com restrições de deslocamentos

Neste item analisa-se novamente a treliça proposta no item anterior (Figura 6.1), porém com a diferença de que agora são impostas restrições de deslocamentos na estrutura. A função de custo a ser otimizada é a mesma do caso anterior, porém as restrições não são apenas de tensões. Para aplicar as restrições de deslocamento utilizou-se o Teorema de Castigliano, possibilitando o relacionamento dos deslocamentos com as áreas, que neste estudo são as variáveis a serem otimizadas.

Como restrições de deslocamentos, foi imposto que o deslocamento máximo dos nós deve ser 0,0508m. O limite superior das variáveis de projeto aumentou para $2,0 \times 10^{-2} \text{m}^2$. As demais restrições, condições e propriedades dos materiais são as mesmas do item anterior e são apresentadas na Tabela 6.1.

A Figura 6.3 mostra a variação da massa da treliça em função do número de iterações. Analisando esta figura pode-se observar que aumentando o número de iterações consegue-se diminuir a massa da estrutura, porém, visualiza-se que a partir das 5000 iterações a solução tende a um valor assintótico, que com 50000 iterações é de 2740kg. Este valor ficou acima do valor apresentado na literatura por Souza (2009), o qual chegou a um valor em torno de 2500kg. A Tabela 6.4 apresenta a comparação das soluções publicadas por Souza (2009) com as obtidas neste trabalho.

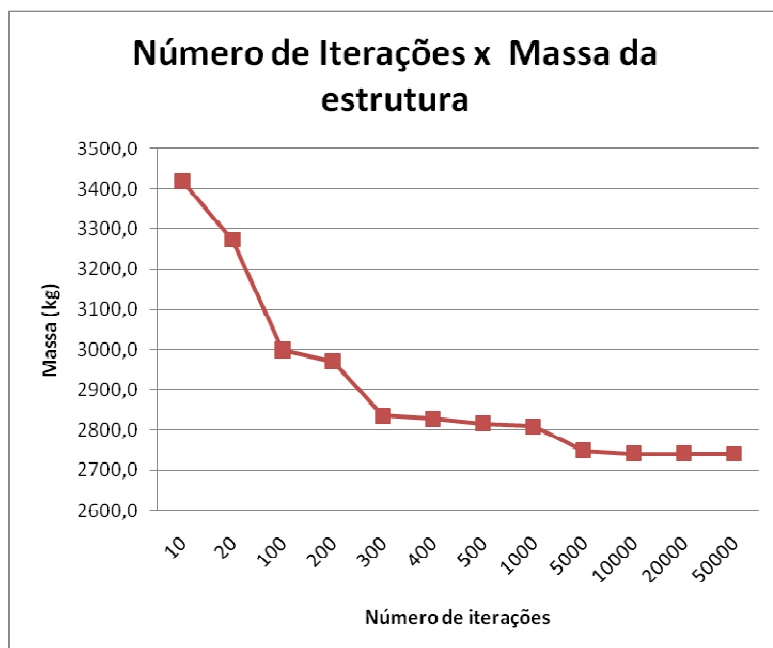


Figura 6.3 – Relação entre o número de iterações e a massa da estrutura.

Tabela 6.4 – Comparação entre soluções.

Variáveis	Algoritmos Genéticos (Haftka,1992)	Harmony Search (presente trabalho)
A1(mm ²)	19692	19135,06
A2(mm ²)	65	4460,38
A3(mm ²)	14969	19029,31
A4(mm ²)	9820	6037,45
A5(mm ²)	65	920,26
A6(mm ²)	355	5038,29
A7(mm ²)	4813	11246,80
A8(mm ²)	13575	11167,43
A9(mm ²)	13891	8662,82
A10(mm ²)	65	6906,53
Massa (kg)	2295	2740

Como se pode observar na Tabela 6.4, neste problema a diferença entre o presente trabalho e o valor encontrado na literatura ficou em torno de 16% a mais de peso na solução proposta pelo *Harmony Search*.

6.3. Trelça tridimensional de 25 barras

A estrutura a seguir é constituída de 25 barras que ligam 10 nós, conforme mostra a Figura 6.4. O conjunto de barras forma uma torre de aproximadamente 5,5 metros de altura e possui restrições de deslocamento nos nós superiores. As restrições, condições de contorno e propriedades do material estão listados na Tabela 6.5, enquanto a Tabela 6.6 fornece os carregamentos aplicados na estrutura.

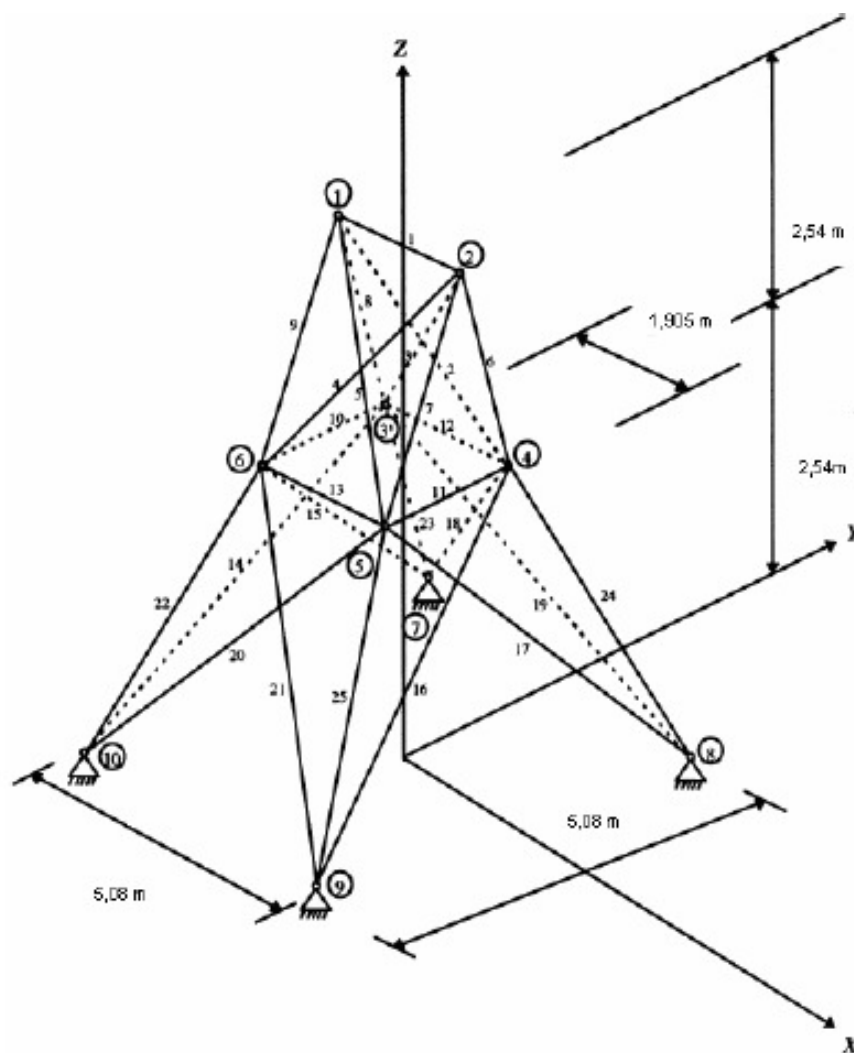


Figura 6.4 – Treliça espacial de 25 barras (Guerra, 2008).

Tabela 6.5 – Dados do problema da treliça de 25 barras.

Módulo de elasticidade	$E = 68,9 \times 10^9 \text{ N/m}^2$
Massa específica	$\rho = 2768 \text{ kg/m}^3$
Tensão de tração e de compressão	$ \sigma < 275 \times 10^6 \text{ N/m}^2$
Deslocamento máximo dos nós	0,00889m
Limite das variáveis de projeto	$6,46 \times 10^{-6} \text{ m}^2 < A_i < 6,46 \times 10^{-3} \text{ m}^2$

Tabela 6.6 – Cargas aplicadas na estrutura.

Nó	Px (kN)	Py (kN)	Pz (kN)
1	4,45	-44,5	-44,5
2	0	-44,5	-44,5
3	2,23	0	0
4	2,67	0	0

A otimização deve ser realizada mantendo-se a relação de simetria da estrutura em relação aos planos x-z e y-z. Por esta razão, 8 variáveis de projeto são usadas para dimensionar os 25 elementos da treliça, de acordo com a Tabela 6.7.

Tabela 6.7 – Variáveis de projeto.

Variáveis de projeto	Conectividades dos nós
1	1-2
2	1-4, 1-5, 2-3, 2-6
3	1-3, 1-6, 2-4, 2-5
4	3-6, 4-5
5	3-4, 5-6
6	3-10, 4-9, 5-8, 6-7
7	3-8, 4-7, 5-10, 6-9
8	3-7, 4-8, 5-9, 6-10

A solução deste problema pode ser vista na Figura 6.5 na qual são apresentadas algumas soluções em relação ao número de iterações. É importante salientar que os outros parâmetros do método foram utilizados sempre os mesmos, visto que após alguns testes se observou que a variação deles piorou a solução dos problemas.

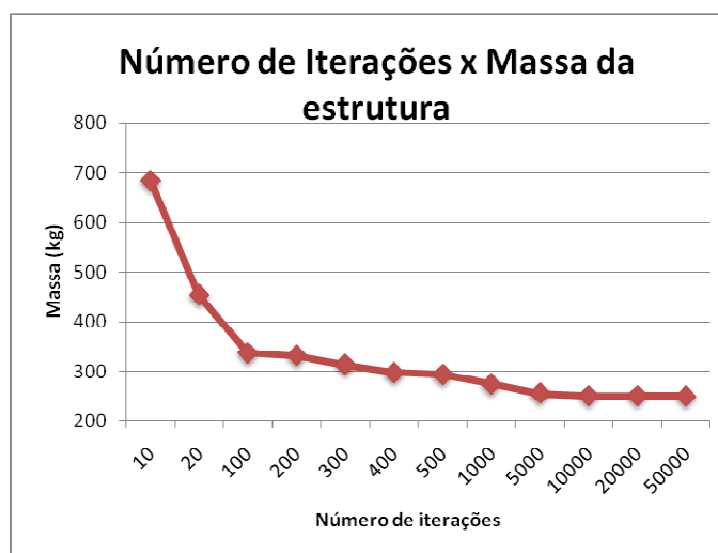


Figura 6.5 – Relação entre número de iterações e a massa obtida.

Observa-se no gráfico da Figura 6.5, que da mesma forma que nos dois casos já apresentados, a solução atinge um valor aproximadamente estável quando se chega a 20000 iterações. No *Harmony Search* o número de iterações é representado pelo número de improvisos que é submetida a função, ou seja, o número de vezes que a solução é recombinada a fim de se obter outra solução.

Na Tabela 6.8 podem-se observar as áreas obtidas com o presente trabalho em relação aos trabalhos já publicados na literatura.

Tabela 6.8 – Comparação entre métodos.

Variáveis	Métodos				
	Zhu (1986)	Rizz (Coello, 1994)	Schimit, (1974)	GA (Erbatur <i>et al.</i> , 1999)	Harmony Search (presente trabalho)
A1(mm ²)	64,51	64,51	64,51	64,51	6448,90
A2(mm ²)	1225,80	1282,57	1267,09	451,61	112,97
A3(mm ²)	1677,41	1929,67	1956,77	129,03	236,40
A4(mm ²)	64,51	6,45	6,45	64,51	65,17
A5(mm ²)	64,51	6,45	6,45	903,22	66,73
A6(mm ²)	516,12	441,28	432,25	709,67	67,51
A7(mm ²)	1354,83	1081,28	1083,86	322,58	104,70
A8(mm ²)	1677,41	1717,41	432,25	2193,54	256,24
Massa (kg)	255,34	247,28	247,31	224,05	249,35

O presente trabalho se mostrou bem coerente, porém a massa não foi a melhor apresentada na literatura já que de quatro métodos apresentados o *Harmony Search* apenas superou o mais antigo que foi testado em 1986, na Figura 6.6 pode-se observar a estrutura estudada com as áreas obtidas na solução do presente trabalho, onde as de mesma cor tem a mesma área.

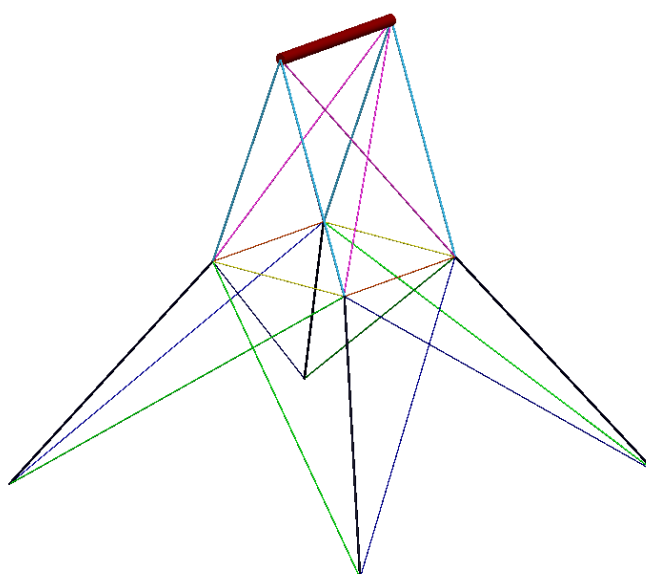


Figura 6.6 – Estrutura com as áreas apresentadas na solução do Harmony Search.

6.4. Treliça tridimensional de 72 barras

O presente exemplo é constituído de uma estrutura de 72 barras e 40 nós que possui a altura aproximada de 12 metros, conforme mostra a Figura 6.7. As restrições e propriedades do material são apresentadas na Tabela 6.9, enquanto a Tabela 6.10 mostra o carregamento atuante na estrutura.

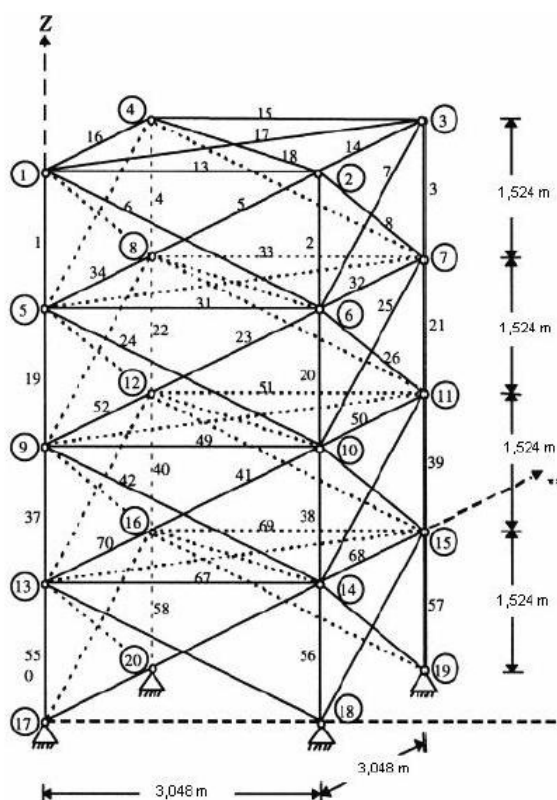


Figura 6.7 – Treliça espacial de 72 barras (Guerra, 2008).

Tabela 6.9 – Dados do problema da treliça de 72 barras.

Módulo de elasticidade	$E = 68,9 \times 10^9 \text{ N/m}^2$
Massa específica	$\rho = 2768 \text{ kg/m}^3$
Tensão de tração e de compressão	$ \sigma < 172 \times 10^6 \text{ N/m}^2$
Deslocamento máximo dos nós	0,00635m
Limite das variáveis de projeto	$3,225 \times 10^{-6} \text{ m}^2 < A_i < 2,580 \times 10^{-3} \text{ m}^2$

Tabela 6.10 – Carga externa.

Nó	P_x (kN)	P_y (kN)	P_z (kN)
1	22,25	22,25	-22,25

A otimização deve ser realizada mantendo-se a relação de simetria da estrutura. Por esta razão, 16 variáveis de projeto são usadas para dimensionar os 72 elementos da treliça, de acordo com a Tabela 6.11.

Tabela 6.11 – Variáveis de projeto.

Áreas	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16
Barras	1-4	5-12	13-16	17-18	19-22	23-30	31-34	35-36	37-40	41-48	49-52	53-54	55-58	59-66	67-70	71-72

A solução deste problema pode ser vista na Figura 6.8 na qual são apresentadas algumas soluções em relação ao número de iterações. Novamente é importante salientar que os outros parâmetros do método foram utilizados sempre os mesmos visto que após alguns testes se observou que a variação deles leva a soluções não tão boas dos problemas.

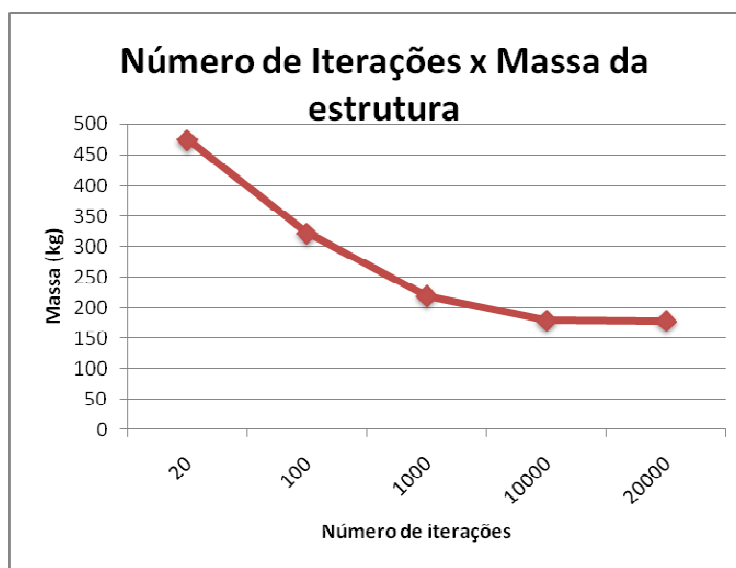


Figura 6.8 – Relação entre número de iterações e a massa obtida.

Como esperado, o gráfico teve um comportamento aproximadamente exponencial, a massa otimizada ficou no valor de 178,53kg. Na Tabela 6.12 podem-se observar as áreas obtidas com o presente trabalho em relação aos trabalhos já publicados na literatura.

Tabela 6.12 – Soluções da literatura e do presente trabalho.

Variáveis	Métodos				
	Venkayya (1971)	Gellatly (1996)	Renwei (1987)	Schimit (1974)	Harmony Search (presente trabalho)
A1(mm ²)	103,87	96,25	105,87	102,25	90,44
A2(mm ²)	359,35	498,90	358,19	382,96	289,90
A3(mm ²)	243,22	292,51	270,12	220,25	362,07
A4(mm ²)	326,45	220,45	371,48	391,99	390,45
A5(mm ²)	394,19	356,19	343,67	170,51	352,21
A6(mm ²)	343,22	392,51	339,09	353,54	332,98
A7(mm ²)	64,51	64,51	64,51	64,51	39,21
A8(mm ²)	64,51	64,51	64,51	97,35	100,57
A9(mm ²)	803,86	660,32	831,80	713,99	634,42
A10(mm ²)	338,06	349,74	335,54	373,74	341,26
A11(mm ²)	64,51	64,51	64,51	64,51	93,51
A12(mm ²)	64,51	64,51	64,51	64,51	58,69
A13(mm ²)	1172,90	944,51	1236,96	1340,90	1094,81
A14(mm ²)	338,06	335,93	335,93	324,77	451,25
A15(mm ²)	64,51	64,51	64,51	64,51	131,34
A16(mm ²)	64,51	64,51	64,51	64,51	34,70
Massa (kg)	172,90	179,61	172,21	176,28	178,53

Novamente, os resultados obtidos no presente trabalho se mostraram coerentes com os valores encontrados na literatura, porém a massa não foi a melhor, superando apenas um resultado comparado.

7. CONCLUSÕES

Após os testes e os resultados obtidos chega-se a conclusão de que o *Harmony Search* é um método de otimização coerente com as soluções já publicadas na literatura mesmo que os valores de massa apresentados no presente trabalho sejam ligeiramente maiores quando comparados com outros métodos de otimização. Entretanto, é importante frisar que algumas soluções apresentadas na literatura quando testadas no *software* Mastan não respeitaram algumas restrições de deslocamento. O método quando testado com diferentes parâmetros apresentou pouca variação a menos do parâmetro denominado quantidade de improvisos, que é diretamente proporcional à otimização. Pode-se observar também que o método é muito menos eficiente que outros apresentados na literatura, pois este necessita de um número altíssimo de iterações para encontrar a solução enquanto outros métodos com bem menos esforço obtém soluções mais eficientes. Conclui-se então que o método não teve bons resultados em relação à outros métodos.

REFERÊNCIAS BIBLIOGRÁFICAS

COELLO, C.A.C., RUDNIK, M., CHRISTIANSEN, A. D. “**Using genetic algorithms for optimal design of trusses**”, Proceedings of the Sixth International Conference on Tools with Artificial Intelligence, New Orlenans, USA, 1994.

ERBATUR, F., HASANÇEBİ, O., TÛTÛNCÛ, I., KILIÇ, H. “**Optimal design of planar and space structures with genetic algorithms**”, Department of civil Engineering, Middle East Technical University, 06531 Ankara Trkey, 1999.

Geem Z.W., Kim J.H., Loganathan G.V. “**A new heuristic optimization algorithm: Harmony Search**”. *Simulation*, 76(2):60–68, 2001.

GELLATLY, R. A. “**Optimal Structural Design**”, AFFFDL-TR-70-165,1971.

Goldberg,D.E. “**Genetic Algorithms in Search, Optimization and Machine Learning**”, Addison-Wesley, Reading, MA, 1989.

Guerra, C. “**Otimizao paramtrica de estruturas treliadas por algoritmos genticos**”. Dissertao submetida ao Corpo Docente do Programa de Ps-Graduao em Engenharia Mecnica, PROMEC, da Escola de Engenharia da Universidade Federal do Rio Grande do Sul, Porto Alegre 2008.

Haftka, R., & Grdal, Z. “**Elements of structural Optimization**”, Kluwer Academic Publishers, 1992.

Holland, J.H. “**Adaptation in natural and artificial systems**”, Univ. of Michigan Press, AnnArbor, 1975.

Kennedy, J., Eberhart, R. C. “**Particle swarm optimization**”. *Proc. IEEE Int'l. Conf. on Neural Networks, IV*, 1942–1948. Piscataway, NJ: IEEE Service Center,1995.

Pham D. T., Ghanbarzadeh A., Koc E., Otri S., Rahim S., M.Zaidi, “**The Bees Algorithm – A Novel Tool for Complex Optimisation Problems**”, in Proceedings of IPROMS 2006 Conference, 2006.

RENWEI, X. , PENG L. “**Estrutural Optimization Based on Second Order Approximations of Functions and Dual Theory. Computer Methods in Applied Mechanics and Engineering**”, vol. 65,pp.101-105, 1987.

Sacalabrin, M.H., Parpinelli R.S., Lopes H.S., “**Paralelizao do algoritmo harmony search utilizando unidade de processamento grfica**”. Mecnica Computacional Vol XXIX, pgs. 7109-7121: Buenos Aires,2010.

SCHMIT, L. A., Farshi, B. W. “**Some Approximation Concepts for Structural Synthesis**”, AIAAJ; vol. 12, pp. 231-234, 1974.

Souza, F. H. **Otimizao de forma e peso de trelias planas e espaciais com restries de tenses e freqncias naturais baseada no algoritmo heurstico de Particle Swarm**. 2010. 20f. Monografia (Trabalho de Concluso do Curso de Engenharia Mecnica) – Departamento de Engenharia Mecnica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2010.

Souza, R. P. “**Otimização de treliças com restrições de falhas combinando técnicas de programação de algoritmos contínuos e discretos**”. Dissertação submetida ao Corpo Docente do Programa de Pós-Graduação em Engenharia Mecânica, PROMEC, da Escola de Engenharia da Universidade Federal do Rio Grande do Sul, Porto Alegre 2009.

VENKAYYA, V. B. “**Design of Optimum Structures**”, Computers & Structures, vol. 1, pp. 265-309, 1971.

Yang, X.S. “**Firefly Algorithm. Nature-Inspired Metaheuristic Algorithms**”. Luniver Press: Cambridge, 2008.

Yang, X.S. “**Nature-Inspired Metaheuristic Algorithms**”. Luniver Press: Cambridge, 2008.

ZHU, D.M. “**An Improved Templeman’s Algorithm for Optimum Design of Truss with Discrete Member Size**”, Engineering Optimization, vol.9, pp. 303-315,1986.

ANEXO I

Algoritmo para a treliça de 10 barras sem a restrição de deslocamento.

```

function [BestGen,BestFitness,gx,MaxItr]=HarmonySearch

% This code has been written with Matlab 7.0
% You can modify the simple constraint handling method using more efficient
% methods. A good review of these methods can be found in:
% Carlos A. Coello Coello,"Theoretical and numerical constraint-handling
techniques used with evolutionary algorithms: a survey of the state of the
art"

clc;
clear;
format long e

global NVAR NG NH MaxItr HMS HMCR PARmin PARmax bwmin bwmax;
global HM NCHV fitness PVB BW gx;
global BestIndex WorstIndex BestFit WorstFit BestGen currentIteration;

NVAR=10;           % number of variables
NG=10;            % number of inequality constraints
NH=0;             % number of equality constraints
MaxItr=50000;     % maximum number of iterations
HMS=4;           % harmony memory size
HMCR=0.9;        % harmony consideration rate  0< HMCR <1
PARmin=0.4;      % minimum pitch adjusting rate
PARmax=0.9;      % maximum pitch adjusting rate
bwmin=0.0001;    % minimum bandwidth
bwmax=1.0;       % maximum bandwidth
PVB=[6.4516e-5 6.4516e-3;6.4516e-5 6.4516e-3;6.4516e-5 6.4516e-3;6.4516e-5
6.4516e-3;6.4516e-5 6.4516e-3;6.4516e-5 6.4516e-3;6.4516e-5 6.4516e-
3;6.4516e-5 6.4516e-3;6.4516e-5 6.4516e-3;6.4516e-5 6.4516e-3]; % range of
variables

% /**** Initiate Matrix ****/
HM=zeros(HMS,NVAR);
NCHV=zeros(1,NVAR);
BestGen=zeros(1,NVAR);
fitness=zeros(1,HMS);
BW=zeros(1,NVAR);
gx=zeros(1,NG);
% warning off MATLAB:m_warning_end_without_block

MainHarmony;

% /*****

function sum=Fitness(sol)

roh=2767.9905;

sum=roh*(9.144*(sol(1)+sol(2)+sol(3)+sol(4)+sol(5)+sol(6))+(12.9316*(sol(7)+s
ol(8)+sol(9)+sol(10))))+eg(sol); % F(x) = f(x) + penalty

```

```

end

% /*****/
function sum=eg(sol)

    % constraints g(x) > 0
    gx(1)=172.37e6 - (869e3/sol(1));
    gx(2)=172.37e6 - (178e3/sol(2));
    gx(3)=172.37e6 - (910e3/sol(3));
    gx(4)=172.37e6 - (266e3/sol(4));
    gx(5)=172.37e6 - (158e3/sol(5));
    gx(6)=172.37e6 - (178e3/sol(6));
    gx(7)=172.37e6 - (658e3/sol(7));
    gx(8)=172.37e6 - (600e3/sol(8));
    gx(9)=517.11e6 - (377e3/sol(9));
    gx(10)=172.37e6 - (252e3/sol(10));

    % we use static penalty function to handle constraints
    sum = 0;
    for i=1:NG
        if(gx(i)<0)
            sum = sum - 1000 * gx(i);
        end
    end
end

% /*****/

function initialize
    % randomly initialize the HM
    for i=1:HMS
        for j=1:NVAR
            HM(i,j)=randval(PVB(j,1),PVB(j,2));
        end
        fitness(i) = Fitness(HM(i,:));
    end
end

% /*****/

function MainHarmony
    % global NVAR NG NH MaxItr HMS HMCR PARmin PARmax bwmin bwmax;
    % global HM NCHV fitness PVB BW gx currentIteration;

    initialize;
    currentIteration = 0;

    while(StopCondition(currentIteration))

        PAR=(PARmax-PARmin)/(MaxItr)*currentIteration+PARmin;
        coef=log(bwmin/bwmax)/MaxItr;
        for pp =1:NVAR
            BW(pp)=bwmax*exp(coef*currentIteration);
        end
        % improvise a new harmony vector
        for i =1:NVAR
            ran = rand(1);

```

```

        if( ran < HMCR ) % memory consideration
            index = randint(1,HMS);
            NCHV(i) = HM(index,i);
            pvbRan = rand(1);
            if( pvbRan < PAR) % pitch adjusting
                pvbRan1 = rand(1);
                result = NCHV(i);
                if( pvbRan1 < 0.5)
                    result =result+ rand(1) * BW(i);
                    if( result < PVB(i,2))
                        NCHV(i) = result;
                    end
                else
                    result =result- rand(1) * BW(i);
                    if( result > PVB(i,1))
                        NCHV(i) = result;
                    end
                end
            end
        else
            NCHV(i) = randval( PVB(i,1), PVB(i,2) ); % random
        end
    selection
    end
    newFitness = Fitness(NCHV);
    UpdateHM( newFitness );

    currentIteration=currentIteration+1;
end
BestFitness = min(fitness);
end
% /*****

function UpdateHM( NewFit )
% global NVAR MaxItr HMS ;
% global HM NCHV BestGen fitness ;
% global BestIndex WorstIndex BestFit WorstFit currentIteration;

if(currentIteration==0)
    BestFit=fitness(1);
    for i = 1:HMS
        if( fitness(i) < BestFit )
            BestFit = fitness(i);
            BestIndex =i;
        end
    end

    WorstFit=fitness(1);
    for i = 1:HMS
        if( fitness(i) > WorstFit )
            WorstFit = fitness(i);
            WorstIndex =i;
        end
    end
end
if (NewFit< WorstFit)

    if( NewFit < BestFit )
        HM(WorstIndex,:)=NCHV;
    end
end

```



```

        BestGen=NCHV;
        fitness(WorstIndex)=NewFit;
        BestIndex=WorstIndex;
    else
        HM(WorstIndex,:)=NCHV;
        fitness(WorstIndex)=NewFit;
    end

    WorstFit=fitness(1);
    WorstIndex =1;
    for i = 1:HMS
        if( fitness(i) > WorstFit )
            WorstFit = fitness(i);
            WorstIndex =i;
        end
    end

    end
end % main if

end %function

% /*****/
function val1=randval(Maxv,Minv)
    val1=rand(1)*(Maxv-Minv)+Minv;
end

function val2=randint(Maxv,Minv)
    val2=round(rand(1)*(Maxv-Minv)+Minv);
end
% /*****/

function val=StopCondition(Itr)
    global MaxItr;
    val=1;
    if(Itr>MaxItr)
        val=0;
    end
end
end

```