



## Métodos de otimização global aplicados na busca dos parâmetros SRC-2D

Carmo\*, L, M, K, e Garabito, G., - CPGf/UFGA/Brasil

\*[klaus@ufpa.br](mailto:klaus@ufpa.br), [german@ufpa.br](mailto:german@ufpa.br)

Copyright 2003, SBGf - Sociedade Brasileira de Geofísica

This paper was prepared for presentation at the 8<sup>th</sup> International Congress of The Brazilian Geophysical Society held in Rio de Janeiro, Brazil, 14-18 September 2003.

Contents of this paper was reviewed by The Technical Committee of The 8<sup>th</sup> International Congress of The Brazilian Geophysical Society and does not necessarily represent any position of the SBGf, its officers or members. Electronic reproduction, or storage of any part of this paper for commercial purposes without the written consent of The Brazilian Geophysical Society is prohibited.

### Abstract

The common-reflection-surface (CRS) stack simulates a zero-offset section from multicoverage seismic data. Whereas conventional reflection imaging methods (e.g. the NMO/DMO stack or prestack migration) require a sufficiently accurate macrovelocity model to yield appropriate results, the CRS stack does not depend on a macrovelocity model. The CRS stacking surface depends on three search parameters: the emergence angle of the ZO ray, as well as two radii of the curvatures. These attributes are associated with the hypothetical normal wave and the normal-incidence-point wave. This paper present a comparative study of global optimization methods - Simulated Annealing, Differential Evolution and Adaptive Cluster Covering - in the problem of determination an optimal parameter triplet for the CRS stacking method.

### Introdução

Tradicionalmente, problemas de otimização são solucionados usando técnicas de otimização linear e não-linear que normalmente assumem que a função minimizada (função-objeto) é conhecida na forma analítica e que tem um único mínimo. Na prática, entretanto, existem muitos problemas que não podem ser descritos analiticamente e muitas funções com múltiplos mínimos. Neste caso é necessário aplicar métodos de otimização global.

O método do empilhamento por superfície de reflexão comum (SRC) produz seções simuladas de afastamento nulo (AN) a partir dos dados de cobertura múltipla. Para meios 2D, o operador de empilhamento SRC depende de três parâmetros que são: o ângulo de emergência do raio central com fonte-receptor nulo ( $\beta_0$ ), o raio de curvatura da onda ponto de incidência normal ( $R_{NIP}$ ) e o raio de curvatura da onda normal ( $R_N$ ). A implementação do método SRC precisa da determinação destes três parâmetros, a partir dos dados sísmicos, por meio de processos de otimização usando como função-objeto a análise de coerência (semblance). A função-objeto deste problema de otimização é multi-modal ou com vários máximos. Neste trabalho descreve-se como um problema de minimização tomando o valor da função de coerência negativo.

Recentemente [2], [3] tem determinado estes parâmetros SRC mediante uma combinação de processos de otimização global e local. O algoritmo utilizado para

otimização global é o Simulated Annealing e para a otimização local é o quasi-Newton. O problema de otimização no método SRC é muito importante para o bom desempenho no que diz respeito a qualidade dos resultados a serem obtidos e principalmente ao custo computacional, comparado com os métodos tradicionalmente usados na indústria sísmica (por exemplo, o empilhamento NMO/DMO). Portanto, existe a necessidade de pesquisar o desempenho dos métodos de otimização global tradicionais e outros, no processo de determinação dos parâmetros do método SRC. Uma comparação de algoritmos de otimização local aplicados na busca dos parâmetros finais que definem a melhor superfície ou operador do empilhamento SRC, foi realizado por Garabito, G, durante a implementação do método SRC. Usando como aproximações iniciais os resultados da otimização global (SA), foram comparados os algoritmos de otimização local Newton-Marquardt (Marquardt, 1963), quasi-Newton (Gill *et al.*, 1981) e Simplex (Nelder and Mead, 1965), tendo melhor desempenho o algoritmo quasi-Newton.

Tomando como base a estratégia de otimização por meio do algoritmo SA [1] aplicada para a implementação do método SRC dada em [4], nós apresentamos a aplicação de mais dois algoritmos de otimização global para a determinação dos parâmetros SRC, estes são: Differential Evolution [8] e Adaptive Cluster Covering [5], [6]. Como resultados preliminares deste trabalho apresentamos os resultados da comparação destes três métodos quanto a eficácia, eficiência e confiabilidade, para determinar os melhores parâmetros SRC. Posteriormente, será implementado o algoritmo de otimização global que tem o melhor desempenho em combinação com um algoritmo de otimização local. Usando como aproximações iniciais os resultados da otimização global dada em Garabito, G., foram realizadas comparações no processo de refinação de dois dos três parâmetros do SRC na otimização local quasi-Newton (Gill *et al.*, 1981), para a busca conjunta dos parâmetros do empilhamento SRC.

### Teoria

#### Empilhamento SRC:

O operador de empilhamento SRC está baseado na determinação de três atributos de frentes de ondas, de duas ondas hipotéticas associadas a um raio de incidência normal. A onda, chamada Ponto de Incidência Normal (onda PIN), propaga-se em forma ascendente a partir de uma fonte no ponto de incidência normal de um refletor em profundidade. A outra onda, chamada Normal (onda N), pode ser obtida a partir de um experimento do refletor explodindo em profundidade. (Hubral, 1983). A expansão hiperbólica em série de Taylor do tempo de trânsito de raios paraxiais a um raio central com incidência normal, expresso em função dos raios de

curvaturas da onda PIN,  $R_{NIP}$ , e da onda N,  $R_N$ , medidas no ponto de emergência do raio central e do ângulo de emergência do raio central,  $\beta_0$ , é dada por:

$$t^2(x_m, h) = \left( t_0 + \frac{2 \operatorname{sen} \beta_0}{v_0} (x_m - x_0) \right)^2 + \frac{2 t_0 \cos^2 \beta_0}{v_0} \left( \frac{(x_m - x_0)^2}{R_N} + \frac{h^2}{R_{NIP}} \right) \quad (1)$$

Na expressão acima  $t_0$  denota o tempo duplo de trânsito do raio normal ligando  $x_0$  e o ponto de reflexão em subsuperfície, e  $v_0$  é a velocidade próxima a superfície que é conhecida a priori. A expressão (1) calcula o tempo de trânsito de um raio com reflexão primária, na vizinhança do raio normal, que tem origem numa fonte na posição  $x_s$  e emerge na posição do receptor  $x_r$  ambos sobre a linha sísmica. O meio-afastamento entre as posições da fonte e do receptor é dado por  $h = (x_r - x_s)/2$  com  $x_m = (x_r + x_s)/2$  denotando o ponto médio comum entre a fonte e o receptor. Conhecendo-se o trio de parâmetros  $(\beta_0, R_{NIP}, R_N)$  para cada ponto de amostragem  $P_o(x_0, t_0)$  da seção AN a ser simulada, com a expressão (1) calcula-se o operador de empilhamento do método SRC no domínio ponto médio – meio afastamento associado a cada ponto de amostragem AN. Portanto, empilhando os eventos sísmicos contidos em cada uma destas superfícies e colocando o resultado no ponto de amostragem AN correspondente, obtém-se a seção de AN simulada pelo método SRC. Na prática, estes três parâmetros que definem o operador SRC são determinados a partir dos dados sísmicos por meio de métodos de otimização.

**Otimização Global dos parâmetros SRC:**

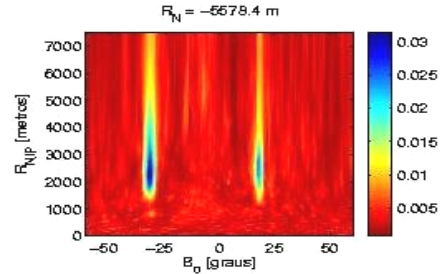
O problema crucial para a implementação do método de empilhamento SRC, consiste na determinação, a partir dos dados sísmicos, dos três parâmetros  $(\beta_0, R_{NIP}, R_N)$  ótimos associados a cada ponto de amostragem da seção AN a ser simulada. A determinação simultânea destes três parâmetros pode ser realizada por meio de processos de otimização utilizando como função-objeto algum critério de coerência. O critério de coerência a ser usado neste trabalho é o semblance (Neidell & Taner, 1997).

Seguindo a estratégia de determinação dos parâmetros apresentada em [2] e [3], onde os três parâmetros são determinados em duas etapas: Na primeira etapa são determinados dois parâmetros  $(\beta_0, R_{NIP})$  e na segunda etapa é determinado o terceiro parâmetro  $(R_N)$ . A redução de (1) para dois parâmetros é possível aplicando a condição de difração  $(R_N = R_{NIP})$ ; por outro lado, no domínio afastamento nulo  $(h = 0)$  a formula (1) também se reduz para dois parâmetros  $(\beta_0, R_N)$ . Isto significa

que na segunda etapa utiliza-se o parâmetro  $\beta_0$  já determinado.

Neste trabalho, os testes de otimização são realizados nos dados de cobertura múltipla de um modelo sintético apresentado em [2] e [3]. Esses dados consistem de 121 seções fonte comum, cada um com 48 traços sísmicos com um tempo de registro de 1.5 s e um intervalo de amostragem de 2 ms.

Como ilustração da função objeto usada nos processos de otimização aplicados ao método SRC, na Figura 1 mostra-se um mapa de coerência no espaço definido pelas coordenadas  $(\beta_0, R_{NIP})$  que foi extraída de um cubo de coerência para um valor fixo de  $R_N$ . Esta função de coerência foi calculada para um ponto de amostragem da seção AN onde existe cruzamento de eventos correspondentes a uma caustica. Observa-se que esta função objeto é multi-modal (vários mínimos), mas com dois mínimos bem definidos correspondentes a dois eventos que se cruzam. Os valores exatos associados ao mínimo global são  $R_{NIP} = 2082$  m,  $\beta_0 = -0.558$  rad. e  $R_N = -5578$ .



**Figura1.** Mapa de coerência para valores de  $\beta_0$  e  $R_{NIP}$ , associados a um ponto de amostragem da seção AN, de um modelo sintético, onde  $R_N$  é fixo.

Em geral, o espaço de busca em três dimensões dos parâmetros SRC é definido pelos seguintes intervalos:  $-50^\circ \leq \beta_0 \leq 50^\circ$ ,  $0 \text{ m} \leq R_{NIP} \leq 15000 \text{ m}$  e  $-30000 \text{ m} \leq R_N \leq +30000 \text{ m}$ .

Neste trabalho, a determinação dos parâmetros do operador SRC será realizada aplicando três métodos de otimização global: Simulated Annealing Differential Evolution e Adaptive Cluster Covering. Serão efetuados testes comparativos referentes aos seguintes indicadores de desempenho:

- **Eficácia:** consiste de como o algoritmo termina chegando ao mínimo (critério de parada).
- **Eficiência:** é uma medida do número de avaliações da função que foram necessárias. Também está relacionado com o tempo de funcionamento do algoritmo.
- **Confiabilidade ou robustez:** é medida pelo número de acertos em achar o mínimo global, ou pelo menos chegar suficientemente próximo.

A seguir é apresentada uma revisão teórica de cada um dos métodos otimização aplicados no problema de otimização do método SRC.

**Algoritmo Simulated Annealing:**

Simulated annealing (SA) é um método de otimização global que imita o processo físico pelo qual um cristal se desenvolve (ou cristaliza) a partir do esfriamento lento (annealing) de um cristal fundido, até atingir o estado de energia mínima. Flutuações aleatórias na energia permitem ao sistema de esfriamento lento escapar de um estado de energia mínima local para alcançar o estado de energia mínima global. Neste trabalho será utilizado o algoritmo Simulated Annealing modificado por [1].

No algoritmo SA o processo de minimização começa com um dado valor inicial de temperatura ( $T = T_0$ ) e, em geral, a partir de um vetor de parâmetros iniciais ( $\mathbf{p} = \mathbf{p}_0$ ) que é gerado aleatoriamente, uma vez que este não tem influência decisiva na determinação final dos parâmetros ótimos. Então, dado um ponto inicial  $\mathbf{p}$  e o respectivo valor da função-objeto  $f(\mathbf{p})$ , a geração de um ponto teste  $\mathbf{p}'$  é realizada por meio da aplicação de uma perturbação em cada elemento de  $\mathbf{p}$ , segundo a relação:

$$p_i' = p_i + \lambda v_i, \quad i = 1, 2, \dots, n \quad (3)$$

onde  $\lambda$  é um número aleatório uniforme distribuído entre  $[-1, 1]$  e  $v_i$  é o  $i$ -ésimo elemento do vetor dos comprimentos do passo  $\mathbf{v}$ . A variável  $n$  representa o número de parâmetros a serem otimizados. Um ponto teste  $\mathbf{p}'$  é aceito incondicionalmente se o valor de  $f(\mathbf{p}')$  for menor que  $f(\mathbf{p})$  (i.e.,  $\Delta f = f(\mathbf{p}') - f(\mathbf{p}) < 0$ ). Então, o ponto  $\mathbf{p}$  é substituído por  $\mathbf{p}'$  e o algoritmo tem um deslocamento descendente. Se o novo  $f(\mathbf{p})$  tiver o menor valor registrado, este valor e o vetor  $\mathbf{p}$  correspondente são guardados como os atuais valores ótimos. Quando o valor  $f(\mathbf{p}')$  do ponto teste tem um valor igual ou maior a  $f(\mathbf{p})$  (i.e.,  $\Delta f \geq 0$ ), um critério de probabilidade (critério de Metropolis) decide se esse ponto teste ascendente é aceito ou não, com base na relação de probabilidade:

$$P = e^{-\Delta f / T} \quad (4)$$

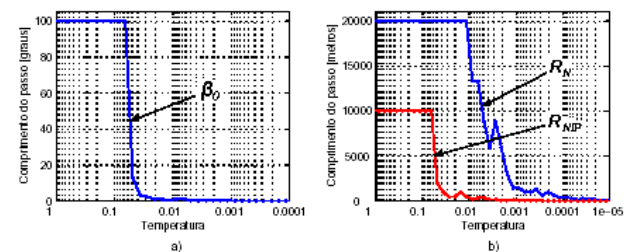
em que  $T$  é a temperatura. Se o novo ponto teste for aceito, então  $\mathbf{p}$  e  $f(\mathbf{p})$  são substituídos por  $\mathbf{p}'$  e  $f(\mathbf{p}')$ , respectivamente, e o algoritmo tem um deslocamento ascendente. Dois fatores diminuem a probabilidade de um deslocamento ascendente ser aceito: valores de temperatura e valores elevados de  $\Delta f$ . Depois de realizar vários ciclos de geração de pontos testes a temperatura  $T$  é diminuída segundo o fator  $r_T$ . Conforme diminui a temperatura os comprimentos dos passos declinam e o algoritmo restringe sua busca na área mais promissora para a otimização, sendo atingido assim o mínimo global. O algoritmo é interrompido quando as diferenças do valor final da função-objeto da temperatura atual com os valores finais da função-objeto das últimas  $N_\varepsilon$  temperaturas e com o valor ótimo da função-objeto são menores do que o valor de tolerância do erro  $\varepsilon$ .

Escolha dos parâmetros do SA:

Outras variáveis que influem na eficiência do algoritmo são  $N_T$ ,  $N_S$  e  $r_T$ . Sendo  $n$  o número de parâmetros, a variável  $N_S$  determina que depois de  $N_S \cdot n$  avaliações da função, o vetor dos comprimentos dos passos é ajustado. A variável  $N_T$  determina que depois de  $N_T \cdot n$  avaliações da função, a temperatura  $T$  é reduzida pelo fator  $r_T$ .

Portanto, uma apropriada seleção da temperatura inicial  $T = T_0$  é crucial para a eficiência do algoritmo SA. Se  $T_0$  for muito baixo os comprimentos dos passos podem ser muito pequenos e a área contendo o mínimo global pode não ser visitada pelo algoritmo. Isto porque a solução atual pode não conseguir escapar do mínimo local. De outra forma, se  $T_0$  for muito alto os comprimentos dos passos podem ser bastante grandes, e uma área ampla demais é visitada, tendo portanto, um elevado número de aceitação dos pontos testes, o que implica num incremento do tempo computacional.

A Figura 2 abaixo, mostra as relações dos comprimentos dos passos com a temperatura para os três parâmetros, utilizando uma temperatura inicial  $T_0 = 1,0$ . Para o parâmetro  $\beta_0$  (Figura 2a) observa-se que os comprimentos dos passos praticamente não mudam para temperaturas maiores que 0,06, porém havendo uma diminuição dos passos para temperaturas menores a 0,06. Esta temperatura é chamada temperatura crítica, a partir da qual começa uma busca eficiente do algoritmo. Na Figura 2b, observa-se que as temperaturas críticas de  $R_{NIP}$  e  $R_N$  são diferentes, cujos valores são aproximadamente 0,06 e 0,01, respectivamente. As temperaturas críticas podem ser diferentes para cada um dos parâmetros, então no caso da otimização simultânea dos três parâmetros a temperatura inicial deve ser estabelecida, tomando como referência o valor máximo das temperaturas críticas, neste caso tomamos como temperatura inicial  $T_0 = 0,1$ .



**Figura 2.** Determinação da temperatura inicial do algoritmo SA para a busca dos parâmetros: a) ângulo de emergência ( $\beta_0$ ) e b) raios de curvatura da onda NIP ( $R_{NIP}$ ) e da onda N ( $R_N$ ).

Com base na análise anterior, foi usada uma temperatura inicial de  $T_0 = 0,1$ , com um fator de decaimento  $r_T = 0,85$ . O valor da tolerância do erro é  $\varepsilon = 0,001$ . No caso da busca é bidimensional tem-se que  $n = 2$  e os valores das variáveis que determinam o número de tentativas associadas a uma certa temperatura são

$N_T = 2$  e  $N_s = 10$ , o que significa que após de 40 avaliações da função a temperatura diminui.

**Algoritmo Differential Evolution:**

Recentemente foi introduzido o algoritmo Differential Evolution (DE) [8]. O algoritmo DE pertence à classe de algoritmos evolucionários, e é um método de busca direta que utiliza  $n_{pop}$  vetores de parâmetros de  $n_{param}$  dimensões,

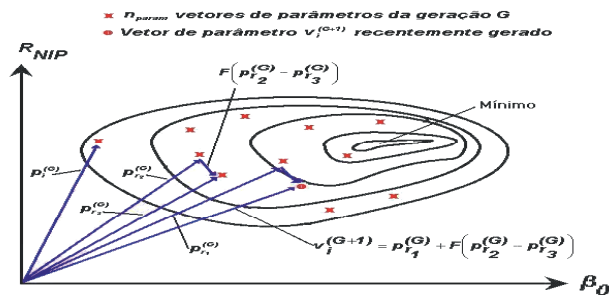
$$p_i^{(G)}, i = 1, \dots, n_{pop} \quad (5)$$

com uma população para cada geração G, isto é, para cada iteração da otimização, onde  $n_{pop}$  não muda durante o processo de minimização. A população inicial é escolhida aleatoriamente e deve tentar cobrir uniformemente todo o espaço de busca. A idéia crucial por traz do DE é o esquema para gerar vetores de parâmetros testes. Basicamente, DE gera novos vetores de parâmetros pela adição da diferença ponderada entre dois vetores da população a um terceiro vetor. Se o vetor resultante produzir um valor menor da função-objeto do que um membro da população predeterminado, o vetor gerado recentemente substituirá o vetor com qual foi comparado na geração seguinte, caso contrário o vetor antigo é conservado. Este princípio básico pode ser estendido para outras variantes do DE, neste trabalho trataremos da variante DE-1 que é abordada a seguir.

Para cada vetor  $p_i^{(G)}, i = 1, \dots, n_{pop}$ , um vetor  $v_i^{(G+1)}$  é gerado conforme

$$v_i^{(G+1)} = p_{r_1}^{(G)} + F(p_{r_2}^{(G)} - p_{r_3}^{(G)}) \quad (6)$$

com  $r_1, r_2, r_3 \in [1, n_{pop}]$ , inteiros e mutuamente diferentes, e  $F > 0$ . Os inteiros  $r_1, r_2$  e  $r_3$  são escolhidos aleatoriamente do intervalo  $[1, n_{pop}]$ . A quantidade  $F \in [0, 2]$  é um fator constante e real que controla a amplificação da variação da diferença  $(p_{r_2}^{(G)} - p_{r_3}^{(G)})$ . O vetor  $p_{r_1}^{(G)}$  que é perturbado para produzir  $v_i^{(G+1)}$  não tem nenhuma relação com  $p_i^{(G)}$  mas é um membro da população escolhido aleatoriamente. O esquema DE-1 especifica que o vetor a ser perturbado é escolhido aleatoriamente e que a perturbação consiste de um vetor diferença ponderada. A Figura 3 ilustra em duas dimensões os diferentes vetores que participam do esquema de geração de vetores.



**Figura 3.** Exemplo 2D de uma função-objeto mostrando suas linhas de contorno e o processo de geração de  $v_i^{(G+1)}$ .

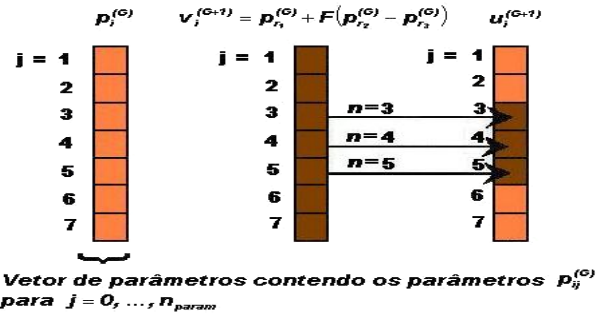
Para aumentar a diversidade do novo vetor de parâmetros, um cruzamento é introduzido. Para este fim, o vetor:

$$u_i^{(G+1)} = (u_{oi}^{(G+1)}, u_{li}^{(G+1)}, K, u_{(n_{param}-1)i}^{(G+1)})$$

com

$$u_{ji}^{(G+1)} = \begin{cases} v_{j,i}^{(G+1)} & \text{para } j = \langle k \rangle_{n_{param}}, \langle k+1 \rangle_{n_{param}}, K, \langle k+L-1 \rangle_{n_{param}} \\ p_{j,i}^{(G)} & \text{para todos os outros } j \in [0, n_{param}-1] \end{cases} \quad (7)$$

é formado. O símbolo  $\langle \rangle_{n_{param}}$  denota a função módulo com modulo  $n_{param}$ . O índice de partida  $k$  em (7) é um inteiro escolhido aleatoriamente do intervalo  $[0, n_{param}-1]$ . O inteiro  $L$ , pertencente a  $[1, n_{param}]$ , denota o número de parâmetros que estão sendo trocados. A relação de probabilidade utilizada neste algoritmo é dada por  $Pr(L = v) = (CR)^{v-1}, v > 0$ , onde  $CR \in [0, 1]$  é a constante de cruzamento e constitui uma variável de controle no desenvolvimento do processo. A decisão aleatória para  $k$  e  $L$  é feita para cada vetor  $u_i^{(G+1)}$  gerado recentemente. A Figura 4 abaixo ilustra o mecanismo do cruzamento DE.



**Vetor de parâmetros contendo os parâmetros  $p_{ji}^{(G)}$  para  $j = 0, \dots, n_{param}$**

**Figura 4.** Ilustração do cruzamento do processo para  $n_{param} = 7, k = 2$  e  $L = 3$ .

Para decidir se deve ou não tornar-se um membro da geração G+1, o novo vetor  $u_i^{(G+1)}$  é comparado com  $p_i^{(G)}$ . Se o vetor  $u_i^{(G+1)}$  produzir um valor da função-objeto menor do que o valor produzido por  $p_i^{(G)}$ , então  $p_i^{(G+1)}$  é substituído por  $u_i^{(G+1)}$ . Caso contrario o vetor antigo  $p_i^{(G)}$  é conservado.

**Escolha dos Parâmetros do DE:**

No DE a capacidade de achar o mínimo global e uma rápida taxa de convergência é muito sensível às variáveis de controle  $NP, F$  e  $CR$ . Algumas regras para a escolha destes parâmetros são dadas abaixo:

- **Tamanho da população NP:** Este valor depende do custo da função-objeto e dos outros parâmetros do algoritmo. Uma grande população tem maior probabilidade de achar o mínimo global para funções multi-modais.
- **Fator de amplificação F:** F não pode ser menor do que um certo valor que impede a convergência prematura. Um grande valor de F aumenta a probabilidade de



escapar de um mínimo local. Uma boa escolha é  $F = 0.6$ . Se suspeitar que com esse valor de  $F$  somente um mínimo local é achado, então deve ser incrementado.

- **Constante de Cruzamento CR:** Para um valor alto de  $CR$  tem-se uma convergência rápida. Entretanto, a partir de um certo valor em diante a velocidade de convergência pode decrescer ou a população pode convergir prematuramente. Este valor depende do custo da função-objeto, uma boa escolha é um valor entre  $CR = 0.3$  e  $CR = 0.9$ .

No presente trabalho foram utilizados uma população  $NP = 50$ , um fator de amplificação  $F = 0.8$  e um cruzamento  $CR = 0.5$ .

#### Algoritmo Adaptive Cluster Covering (ACCO):

O algoritmo de busca aleatória ACCO [5], [6] é um algoritmo que se caracteriza pela combinação de processos de redução, agrupamento e cobertura, aplicados para busca de parâmetros em geral.

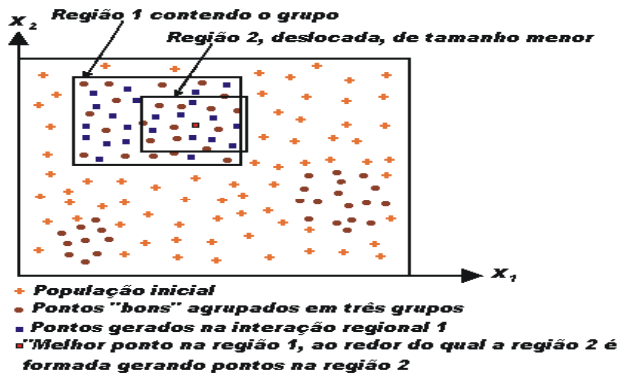


Figura 5. Interações no algoritmo ACCO no caso 2D.

A estratégia ACCO é baseada nos seguintes princípios:

- **Agrupamento:** significa a identificação dos grupos de pontos mutuamente próximos no espaço de busca e é utilizado para identificar o subdomínio mais promissor no qual continua a busca global pela cobertura ativa do espaço.
- **Cobertura contraindo subdomínios:** Cada subdomínio é coberto aleatoriamente, em geral, os valores da função-objeto são avaliados em pontos gerados por uma distribuição uniforme. A cobertura é repetida várias vezes nos subdomínios que são progressivamente reduzidos de tamanho.
- **Adaptação:** Algoritmos adaptáveis atualizam seu comportamento algorítmico dependendo de nova informação revelada sobre o problema. No ACCO a adaptação consiste em deslocar a sub-região de busca, encolhendo-a, e mudando a densidade (número de pontos) de cada cobertura – dependendo da avaliação anterior do minimizador global.
- **Processo aleatório periódico:** Devido ao caráter probabilístico da geração dos pontos, qualquer estratégia de busca aleatória pode simplesmente perder uma região promissora de busca (similar à situação em que procedimentos de buscas locais podem perder o mínimo global). Assim para reduzir esse perigo, a população inicial sofre novamente

processo aleatório, isto é, resolver o problema várias vezes e/ou aplicar um novo processo aleatório em algumas populações nas etapas intermediárias.

#### Escolha dos parâmetros do ACCO:

Os intervalos de valores dos parâmetros utilizados neste algoritmo são os seguintes: população inicial  $N = 30, \dots, 1000$ ; população inicial reduzida  $p = 40, \dots, 300$ ; número de grupos dependendo de  $p$ ,  $K_N = 3, \dots, 20$ ; tamanho inicial  $r_k$  de uma amostra de grupo variável partindo do número inicial de pontos no grupo; número  $S_k$  de pontos para o qual  $r_k$  é reduzido para  $r_k$ ;  $u = 30\%$  porcentagem dos melhores pontos para o critério de parada  $C_2$ ; número de interações regionais  $e_1 = 3, \dots, 15$ ; número mínimo de interações regionais feitas sem melhoramentos no valor da função:  $e_2 = 2, \dots, 5$ ;  $w = 1\%, \dots, 5\%$ ;  $v = 70\%, \dots, 95\%$ ;  $T = 1, \dots, 5$ . Um parâmetro crítico apresentado é o número de interações regionais  $e_1$ . Quando tem um valor muito baixo impede de ser alcançado o mínimo global, e quando tem um valor muito alto o número de avaliações da função incrementa conduzindo a um mínimo que é normalmente atingido nas etapas subseqüentes. Foi achada a seguinte regra razoável para ser usada: para  $2 \leq n \leq 4$ ,  $e_1$  é toma o valor 3, e para grande valor de  $n$ ,  $e_1$  toma o valor de  $n/10 + 4$ . Um parâmetro importante é o numero de pontos gerados  $r_k$  em cada iteração, que normalmente é atribuído o numero original de pontos no grupo.

Outros parâmetros críticos são o tamanho da população inicial ( $N$ ) e o número de pontos agrupados ( $p$ ). Decidiu-se manter a dependência linear entre  $N$  e a dimensão  $n$ , e entre  $p$  e  $n$ . Para comparação a seguinte regra pode ser usado: com  $n$  de 2 a 30,  $N$  aumentando de 50 a 300, e  $p$  de 40 a 200 respectivamente. Neste trabalho, foram utilizados os seguintes valores:  $N = 150$ ,  $p = 100$ ,  $K_N = 3$ ,  $e_1 = 5$ ,  $e_2 = 3$ .

#### Resultados

A comparação de algoritmos propostos por autores diferentes é sempre uma tarefa difícil porque depende muito dos detalhes de execução dos programas. Portanto a seguir, apesar das diferenças de parâmetros e condições de parada, foi feito um esforço de realizar a comparação na determinação simultânea, inicialmente, de três parâmetros  $(\beta_0, R_{NIP}, R_N)$  e, posteriormente, na determinação de dois parâmetros  $(\beta_0, R_{NIP})$ .

Os gráficos das Figuras 6 e 7 mostram o progresso da minimização. Os resultados são uma média de vários testes dos três algoritmos que são mostrados nas Figuras 6 e 7. Nestas figuras são mostradas curvas com os valores da função-objeto (eixo vertical) com relação ao número de avaliações da função-objeto (eixo horizontal). Os resultados da comparação podem ser brevemente resumidos como segue.

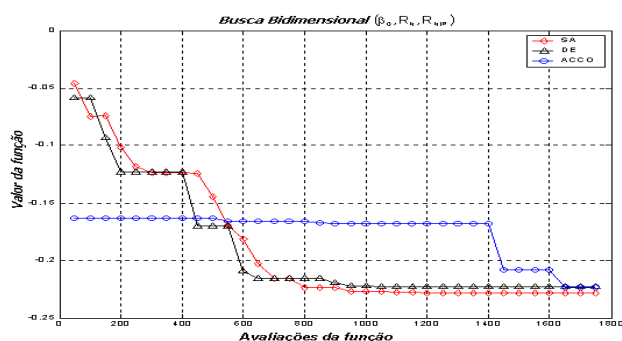


Figura 6. Média do desempenho dos algoritmos para busca de três parâmetros.

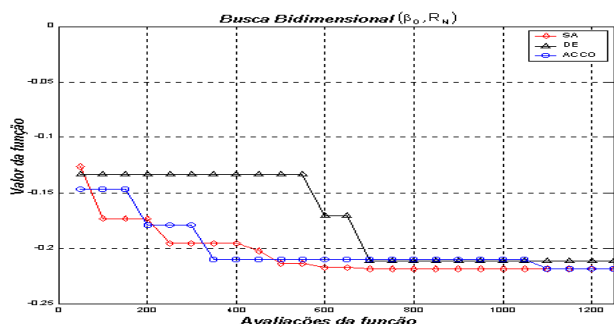


Figura 7. Média do desempenho dos algoritmos para busca de dois parâmetros.

Para o problema de otimização abordado neste trabalho os algoritmos Simulated Annealing (SA) e Differential Evolution (DE), mostram-se mais eficientes, isto é, são mais rápidos em encontrar o valor mínimo da função-objeto ( $S = -0.218$ ) para a busca dos três parâmetros (Figura 6), enquanto que o algoritmo Adaptive Cluster Covering (ACCO) não teve bom desempenho para esta busca. Já na Figura 7, que ilustra a busca de dois parâmetros, observa-se uma melhora considerável no desempenho do algoritmo ACCO, que juntamente com SA mostrou-se mais eficiente do que o DE, já que este último demora a encontrar o mínimo.

A confiabilidade pode ser medida pelo número de êxitos em achar o mínimo global com exatidão pré-definida. Por causa do caráter aleatório da busca, nenhum algoritmo pode ser 100% de confiança. Portanto, como mostrado na Figura 6 o algoritmo ACCO não é de muita confiança para a busca de três parâmetros, enquanto que os algoritmos SA e DE podem ser considerados de confiança para esta busca. Por outro lado, observando a Figura 7 podemos dizer que os três algoritmos são considerados de confiança para a busca de dois parâmetros.

### Conclusões

O algoritmo DE mostrou ter boa eficiência, eficácia e confiabilidade na busca simultânea do trio de parâmetros do operador SRC. Mas mostrou uma queda de eficiência para a busca de dois parâmetros.

O algoritmo ACCO apresentando um baixo desempenho na busca simultânea dos três parâmetros. Já para a busca de dois parâmetros apresenta boa eficiência, eficácia e confiabilidade.

O algoritmo SA apresenta o melhor comportamento entre os três algoritmos mostrando boa eficiência, eficácia e confiabilidade tanto na busca de dois parâmetros como na busca do trio de parâmetro simultaneamente.

É válido mencionar que o número total de avaliações da função-objeto e o equilíbrio entre a eficácia e eficiência, depende significativamente dos vários parâmetros existentes em cada algoritmo, e são definidos pelo usuário. Portanto, é necessário que façamos mais testes e um melhor ajuste de parâmetros para obtermos melhor desempenho algorítmico.

### Agradecimentos

Agradecemos ao Conselho Nacional de Desenvolvimento Tecnológico (CNPq-Brasil) pelo suporte financeiro e ao Senhor Paolo Marchetti da AGIP (Divisão ENI), Milan (Itália), pela permissão de utilizarmos o programa do algoritmo ACCO.

### Referências

- [1] Corana, A.; Marchesi, M.; Martini, C. Ridela, S. 1987. Minimizig multimodal functions of continuous variables with 'Simulated Annealing' algorithm. *ACM Transactions on Mathematical Software*, 13(3): 262-280.
- [2] Garabito, G., Cruz, J.C.R., Hubral, P., and Costa, J. 2001a. Common reflection surface stack by global optimization: 71th Annual Internat. Mtg., Soc. Expl. Geophys. Expanded Abstracts.
- [3] Garabito, G., Cruz, J. C. R., Hubral, P. and Costa, J., 2001b. Empilhamento de superfícies de reflexão comum com mergulhos conflitantes. *Resumo Expandido*, 7º Congresso Internacional da Sociedade Brasileira de Geofísica.
- [4] Garabito, G. 2001. Empilhamento Sísmico Por Superfície de Reflexão Comum: Um novo algoritmo usando otimização global e local (in portuguese): PhD thesis, Federal University of Para (Brazil).
- [5] Solomatine, D.P., 1999a, Two Strategies of Adaptive Cluster Covering with Descent and Their Comparison to Other Algorithms. *Journal of Optimization Global*, vol. 14, Nº 1, pp. 55-78.
- [6] Solomatine, D.P., 1999b, Random search methods in model calibration and pipe network design. *Water Industry Systems: Modelling and Optimization Applications*, vol. 2. Research Studies Press, Ltd. Baldock, UK, pp. 317-332.
- [7] Solomatine, D.P., 1998. Genetic and other global optimization algorithms – comparison and use in calibration problems. *Proc. Int. Conf. Hydroinformatics – 98*, pp. 1021-1027.
- [8] Storn, R. and Price, K. 1996, "Minimizig the real functions of the ICEC' 96 contest by Differential Evolution" *IEEE Conference on Evolutionary Computation*, Nagoya, pp. 842-844.
- [9] Storn, R. and Price, K. 1995, "Differential Evolution – A simple and efficient adaptive scheme for global optimization over continuous spaces", Technical Report TR-95-012. Disponível em: <http://citeseer.nj.nec.com/182432.html>