

## NOVAS VARIANTES PARA O MÉTODO DE PENALIZAÇÃO ADAPTATIVO (APM) PARA PROBLEMAS DE OTIMIZAÇÃO COM RESTRIÇÕES

### Rafael de Paula Garcia

Universidade Federal de Juiz de Fora  
Rua José Lourenço Kelmer, s/n. São Pedro, 36036-330. Juiz de Fora - MG.  
garcia816@gmail.com

### Bernardo Galdeano Montá

Universidade Federal de Juiz de Fora  
Rua José Lourenço Kelmer, s/n. São Pedro, 36036-330. Juiz de Fora - MG.  
b.monta@terra.com.br

### Érica da Costa Reis Carvalho

Universidade Federal de Juiz de Fora  
Rua José Lourenço Kelmer, s/n. São Pedro, 36036-330. Juiz de Fora - MG.  
ericacrcunipac@gmail.com

### Helio José Corrêa Barbosa

LNCC/MCT  
Rua Getúlio Vargas, 333. 25651-070. Petrópolis - RJ.  
hcbm@lncc.br

### Heder Soares Bernardino

Universidade Federal de Juiz de Fora  
Rua José Lourenço Kelmer, s/n. São Pedro, 36036-330. Juiz de Fora - MG.  
heder@ice.ufjf.br

### Afonso Celso de Castro Lemonge

Universidade Federal de Juiz de Fora  
Rua José Lourenço Kelmer, s/n. São Pedro, 36036-330. Juiz de Fora - MG.  
afonso.lemonge@ufjf.edu.br

## RESUMO

Metaheurísticas inspiradas na natureza são normalmente desenvolvidas para resolver problemas de otimização sem restrições, onde se busca o conjunto de variáveis de decisão em um espaço de busca, que minimiza (ou maximiza) uma função objetivo. Os problemas práticos de otimização muitas vezes envolvem várias restrições e a aplicação direta dessas metaheurísticas não é possível sem alterações. É comum considerar técnicas para o tratamento de restrições. Estas técnicas podem ser classificadas como diretas (viáveis ou interiores), quando apenas os elementos viáveis no espaço de busca são considerados, ou como indiretas (exteriores), quando ambos os elementos, viáveis e inviáveis, são utilizados durante o processo de busca.

O Método de Penalização Adaptativo (*Adaptive Penalty Method*, ou simplesmente APM) considerado neste trabalho adapta o valor dos coeficientes de penalização de cada restrição fazendo uso em informações da população, tais como a média da função objetivo e o nível de violação de cada restrição.

Algumas variações deste método foram propostas recentemente e podem ser resumidas como: i) *Sporadic APM* com a acumulação das violações das restrições. A segunda variante

é semelhante à primeira, mas as violações das restrições não são acumuladas. Uma terceira variante, chamada APM com coeficientes de penalização monotônicos, é semelhante ao APM original, em que os coeficientes de penalização são adaptados em cada geração, mas nenhum deles pode ter seu valor reduzido ao longo do processo evolucionário. Finalmente, o *dumping* APM é a quarta variante e usa uma média ponderada entre os valores atuais e anteriores dos coeficientes de penalização e a violação das restrições não são acumuladas durante a evolução.

Novas variantes do APM aplicados a problemas de otimização com restrições são propostos neste trabalho, a fim de tentar melhorar o APM original analisando o desempenho dos mesmos. O desempenho destas novas variantes são examinadas usando problemas testes a partir das referências de computação evolucionária amplamente discutidas na literatura.

**PALAVRAS CHAVE: métodos de penalização, algoritmos genéticos, APM. Área principal: MH - Metaheurísticas.**

### ABSTRACT

Nature inspired metaheuristics are usually designed to solve unconstrained optimization problems, where one seeks for set of design variables into to a search space, which minimizes (or maximizes) an objective function. In general, practical optimization problems often involve several constraints and the straightforward application of these metaheuristics is not possible without changes. It is common to consider techniques for handling constraints. These techniques can be classified as direct (feasible or interior), when only feasible elements in the search space are considered or as indirect (exterior), when both feasible and infeasible elements are used during the search process.

The Adaptive Penalty Method (APM) considered in this paper was proposed by Barbosa & Lemonge (2004), adaptively sizes the penalty coefficient of each constraint using information from the population such as the average of the objective function and the level of violation of each constraint.

Variants of this method were proposed recently and they can be summarized as: i) the sporadic APM with constraint violation accumulation. A second variant is similar to the first one except in the item (i), where the constraints violations are not accumulated. A third variant, called APM with monotonic penalty coefficients, is similar to the original APM, where the penalty coefficients are adapted at each generation, but none of them is allowed to have its value reduced along the evolutionary process. Finally, the APM with “dumping” is the fourth variant that uses a weighted average between the current and previous values of the penalty coefficients and the constraint violation are not accumulated.

New variants of APM applied to constrained optimization problems are proposed in this paper, in order to try to improve the original APM analysing their performances. For example, the values of the worst (feasible or infeasible) and the worst feasible are considered in the evaluation of the penalty coefficients. The performance of these new variants are examined using test problems from the evolutionary computation references.

**KEYWORDS: penalty methods; genetic algorithms; APM. Área principal: MH - Metaheurísticas.**

## 1. Introdução

Meta-heurísticas bio-inspiradas são normalmente concebidas tendo em mente os problemas de otimização sem restrições, em que se procura, para um conjunto de variáveis  $x$ , pertencentes a um espaço de busca, a minimização (ou maximização) de uma função real  $f(x)$ . No entanto, problemas reais de otimização muitas vezes envolvem restrições e a aplicação direta destes tipos de algoritmos não é possível.

As técnicas para lidar com restrições no âmbito das meta-heurísticas bio-inspiradas podem ser classificadas como diretas (interiores), quando apenas elementos viáveis pertencentes ao espaço de busca são considerados, ou como indiretas (exteriores), quando ambos os elementos viáveis e inviáveis são utilizados durante o processo de busca. As técnicas diretas incluem o uso de: a) operadores que geram sempre novas soluções viáveis, b) decodificadores especiais, c) técnicas de reparo e d) “pena de morte”, quando qualquer elemento inviável é simplesmente descartado do conjunto de soluções candidatas. Por outro lado, as técnicas indiretas compreendem o uso de: a) multiplicadores de Lagrange b) função aptidão bem como valores das violações das restrições dentro de uma otimização multiobjetivo c) técnicas específicas de seleção de soluções, e d) para qualquer indivíduo inviável atribui-se uma penalização à sua aptidão.

Existe um considerável número de publicações na literatura no contexto destas técnicas. Várias destas técnicas podem ser encontradas nas referências [1-13]. Aquelas que necessitam do conhecimento das restrições de forma explícita, têm a aplicação limitada perante a grande maioria dos problemas, principalmente os oriundos de áreas de engenharia.

Para o tratamento de problemas de otimização com restrições, explícitas ou não, é comum o uso de técnicas de penalização que, em geral, são de fácil implementação. Barbosa & Lemonge [14] propuseram um método de penalização adaptativa chamado *Adaptive Penalty Method* (APM) para o tratamento de problemas de otimização com restrições. O APM não demanda o conhecimento da forma explícita das restrições como funções das variáveis do problema e é livre de parâmetros a serem definidos pelo usuário.

Diferentemente de outras estratégias de penalização, onde um único parâmetro de penalização é usado para todas as restrições, o APM define automaticamente o parâmetro de penalização correspondente a cada restrição ao longo do processo evolutivo. O APM foi idealizado para um algoritmo genético (AG) do tipo geracional, onde a ideia principal do método está na definição dos parâmetros de penalização que são mais ou menos rigorosos, de acordo com o grau de dificuldade de cada restrição.

Em [15], Barbosa & Lemonge propuseram 4 variantes do APM. A primeira variante calcula as violações como no método original mas, as atualizações de todos os coeficientes de penalização, para cada restrição, ocorre após um determinado número de gerações definido pelo usuário. A segunda variante, chamada *Sporadic APM*, acumula os valores das violações de cada restrição durante um determinado número de gerações e então atualiza os parâmetros de penalização com estes valores acumulados. O APM com coeficientes de penalização determinados de forma monotônica corresponde à terceira variante, em que os coeficientes de penalização são calculados como no método original, porém nenhum desses coeficientes pode ser reduzido ao longo do processo evolutivo. Finalmente, pode-se pensar em reduzir as oscilações dos coeficientes de penalização através de uma média ponderada entre o valor atual de um coeficiente e o novo valor previsto pelo método. Esta variante é chamada *Damping APM*.

Neste artigo propõe-se novas variantes do APM que são derivações destas propostas em [15]. Para estas novas variantes será apresentado um estudo comparativo e detalhado com

o objetivo de identificar qual a variante possui o melhor desempenho em um conjunto de 24 funções [16] amplamente testadas na literatura.

Este conjunto de problemas apresenta variáveis contínuas e é conhecido como *G-Suíte*. Os perfis de desempenho propostos por [17] foram adotados como meio de visualizar e comparar os resultados obtidos.

Na seção seguinte, encontra-se uma breve descrição dos principais métodos de penalização. Na seção 3 são apresentadas algumas modificações para o método auto-adaptativo, APM, e na seção 4 são apresentados experimentos numéricos bem como seus resultados. O artigo termina com algumas conclusões.

## 2. Métodos de Penalização

Um problema de otimização com restrições em  $\mathbb{R}^n$  pode ser definido:

$$\min f(\vec{x}), \quad \vec{x} = [x_1, x_2, \dots, x_n]$$

sujeito à:

$$g_i(\vec{x}) \leq 0, \quad i = 1, \dots, q \quad (1)$$

$$h_j(\vec{x}) = 0, \quad j = q + 1, \dots, m \quad (2)$$

$$x_k^L \leq x_k \leq x_k^U, \quad k = 1, \dots, n. \quad (3)$$

Onde  $g_i(\vec{x})$  são as restrições de desigualdade e  $h_i(\vec{x})$  as de igualdade. Comumente, restrições de igualdade são transformadas em inequações da seguinte forma:

$$|h_j(\vec{x})| - \epsilon \leq 0, \quad j = q + 1, \dots, m \quad (4)$$

A solução  $\vec{x}$  é considerada como factível se  $g_i(\vec{x}) \leq 0$ , para  $i = 1, \dots, q$  e  $|h_j(\vec{x})| - \epsilon \leq 0$ , para  $j = q + 1, \dots, m$ .  $\epsilon$  é uma tolerância admissível nas restrições de igualdade e assume valores pequenos. Neste trabalho,  $\epsilon$  assume valor igual a  $10^{-4}$ .

Técnicas de penalização podem ser classificadas como *multiplicativas* ou *aditivas*. No primeiro caso, um fator positivo de penalização é utilizado a fim de aumentar o valor da função objetivo dos indivíduos inactíveis em problemas de minimização. No caso aditivo, uma função de penalização é adicionada à função objetivo dos indivíduos inactíveis.

O método mais usado para esta última abordagem é:

$$F(\vec{x}) = f(\vec{x}) + kP(\vec{x}) \quad (5)$$

tal que

$$P(\vec{x}) = \begin{cases} 0, & \text{se } \vec{x} \text{ factível} \\ k \sum_{j=1}^m (v_j(\vec{x}))^\beta, & \text{caso contrário} \end{cases} \quad (6)$$

onde  $F(\vec{x})$  é conhecida como função aptidão ou função "*fitness*", que retorna o valor da avaliação de uma solução candidata em um algoritmo populacional bio-inspirado.  $k$  é o parâmetro de penalização e  $\beta$  é usualmente definido com o valor igual a 2. O valor desse parâmetro de penalização  $k$  é definido por um processo demorado de tentativa e erro, introduzindo complexidade na busca.

Neste artigo será usado um método para tratamento de restrições proposto em [19] para a realização dos estudos comparativos dos resultados. Este método é descrito como: os indivíduos são ordenados de maneira que qualquer solução factível é preferida à qualquer solução inactível;

entre duas soluções factíveis, a que possuir o melhor valor de função objetivo é preferida; entre duas soluções infactíveis, a que violar menos é preferida. O esquema é apresentado da seguinte maneira:

$$F(\vec{x}) = \begin{cases} f(\vec{x}), & \text{se } \vec{x} \text{ factível} \\ f_{max} + \sum_{j=1}^m v_j(\vec{x}), & \text{caso contrário} \end{cases} \quad (7)$$

tal que  $f_{max}$  é o valor da função objetivo do pior indivíduo factível.

A maior vantagem desse método é que não é necessário nenhum parâmetro para manipular as restrições.

O método APM, usado como base para as novas variantes propostas neste artigo, tem a função aptidão definida como:

$$F(\vec{x}) = \begin{cases} f(\vec{x}), & \text{se } \vec{x} \text{ é factível} \\ \bar{f}(\vec{x}) + \sum_{j=1}^m k_j v_j(\vec{x}), & \text{caso contrário} \end{cases} \quad (8)$$

tal que

$$\bar{f}(\vec{x}) = \begin{cases} f(\vec{x}), & \text{se } f(\vec{x}) > \langle f(\vec{x}) \rangle \\ \langle f(\vec{x}) \rangle, & \text{se } f(\vec{x}) \leq \langle f(\vec{x}) \rangle \end{cases} \quad (9)$$

e  $\langle f(\vec{x}) \rangle$  é a média dos valores da função objetivo da população atual.

O parâmetro de penalização  $k_j$  é definido como:

$$k_j = |\langle f(\vec{x}) \rangle| \frac{\langle v_j(\vec{x}) \rangle}{\sum_{l=1}^m [\langle v_l(\vec{x}) \rangle]^2} \quad (10)$$

tal que  $\langle v_j(\vec{x}) \rangle$  é o somatório das violações  $j$  de todos os indivíduos dividido pelo número de indivíduos da população (factíveis e infactíveis).

### 3. Variantes para o APM

Barbosa & Lemonge [18] apresentaram algumas variações para o APM, que são descritas a seguir:

- *Sporadic* APM (APM.Spor): o método (i) calcula as violações das restrições  $v_j$  na população atual, (ii) atualiza os coeficientes de penalização  $k_j$  e (iii) os mantém por uma quantidade fixa de gerações;
- *Sporadic Accumulation* APM (APM.Spor.Acum): o método (i) acumula as violações das restrições por um número fixo de gerações, (ii) atualiza os coeficientes de penalização  $k_j$  e (iii) os mantém por uma quantidade fixa de gerações;
- *Monotonic* APM (APM.Mono): o método funciona da mesma maneira que o APM original, porém se  $k_j^{(t)} < k_j^{(t-1)}$  então  $k_j^{(t)} = k_j^{(t-1)}$ , no qual  $t$  equivale à atual geração em execução.
- *Sporadic Monotonic* APM (APM.Mono.f): funciona da mesma maneira que o *Monotonic*, mas atualiza os coeficientes de penalização  $k_j$  e os mantém por um número fixo de gerações;
- *Damping* APM (APM.Damp): o método é definido como  $k_j^{(t)} = \theta k_j^{(t)} + (1 - \theta) k_j^{(t-1)}$  tal que  $\theta \in [0, 1]$  e,  $t$ , equivale à atual geração em execução.

Neste artigo são apresentadas novas variantes para o APM. Estas variantes são descritas como:

- Variante 01:

$$\bar{f}(x) = \begin{cases} f(\vec{x}), & \text{se } f(\vec{x}) > \lfloor f(\vec{x}) \rfloor \\ \lfloor f(\vec{x}) \rfloor, & \text{se } f(\vec{x}) \leq \lfloor f(\vec{x}) \rfloor \end{cases} \quad (11)$$

tal que  $\lfloor f(\vec{x}) \rfloor$  é o pior indivíduo factível da função objetivo da população atual. Caso não existam indivíduos factíveis, é utilizada a média da função objetivo;

- Variante 02:

$$k_j = \lfloor f(\vec{x}) \rfloor \left| \frac{\langle v_j(\vec{x}) \rangle}{\sum_{l=1}^m [\langle v_l(\vec{x}) \rangle]^2} \right| \quad (12)$$

tal que  $\lfloor f(\vec{x}) \rfloor$  é o pior indivíduo factível da função objetivo da população atual. Caso ele não exista, é utilizada a média da função objetivo;

- Variante 03:  $\langle v_j(\vec{x}) \rangle$  é o somatório das violações de todos os indivíduos na restrição  $j$  dividido pelo número de indivíduos que violam essa restrição;
- Variante 04:

$$k_j = \langle f(\vec{x}) \rangle \left| \frac{\langle v_j(\vec{x}) \rangle}{\sum_{l=1}^m [\langle v_l(\vec{x}) \rangle]^2} \right| \quad (13)$$

tal que  $\langle f(\vec{x}) \rangle$ , que antes representava a média da função objetivo, agora representa o somatório da função objetivo de todos os indivíduos da população atual dividido pelo número de indivíduos infactíveis;

- Variante 05:

$$\bar{f}(x) = \begin{cases} f(\vec{x}), & \text{se } f(\vec{x}) > \langle f(\vec{x}) \rangle \\ \langle f(\vec{x}) \rangle, & \text{se } f(\vec{x}) \leq \langle f(\vec{x}) \rangle \end{cases} \quad (14)$$

tal que  $\langle f(\vec{x}) \rangle$ , que antes representava a média da função objetivo, agora representa o somatório da função objetivo de todos os indivíduos da população atual dividido pelo número indivíduos infactíveis.

Na Tabela 1, são apresentadas as novas variantes desenvolvidas, nomeadas na primeira coluna, bem como algumas combinações entre elas. Portanto, quando referir-se ao método APM\_Med.3, por exemplo, será considerada as seguintes modificações no método APM original:

- No cálculo do parâmetro de penalização  $k_j$ ,  $\langle v_j(\vec{x}) \rangle$  será o somatório das violações de todos os indivíduos na restrição  $j$  dividido pelo **número de indivíduos que violam** essa restrição;
- Ainda no cálculo do parâmetro de penalização  $k_j$ ,  $\langle f(\vec{x}) \rangle$ , que antes representava a média da função objetivo, agora representa o somatório da função objetivo de todos os indivíduos da população atual dividido pelo **número de indivíduos infactíveis**.

Métodos	Variantes				
	01	02	03	04	05
APM_worst	X				
APM_worst_2		X			
APM_worst_3	X	X			
APM_Med			X		
APM_Med_2			X	X	X
APM_Med_3			X	X	
APM_Med_4			X		X
APM_Med_5					X
APM_Med_6				X	
APM_Med_7				X	X
APM_Med_Worst	X		X		
APM_Med_Worst_2	X	X	X		

Tabela 1: Novas variantes para o APM.

#### 4. Experimentos Computacionais

Foi implementado um Algoritmo Genético geracional em linguagem de programação C, com codificação Gray das variáveis, seleção por torneio e elitismo (o melhor elemento factível é copiado para a geração seguinte juntamente com uma cópia a qual somente 1 bit é trocado aleatoriamente). Todos os testes utilizaram o mesmo conjunto de parâmetros: 50 indivíduos para o tamanho da população, 5000 gerações, mutação com taxa de 4%, crossover uniforme com probabilidade igual a 90% e 25 bits para cada variável.

Um conjunto de problemas teste com 24 funções (G1-G24) testadas largamente na literatura, denominado *G-Suíte* e propostas por Liang et al. [16], foi utilizado para analisar a eficiência de cada método. O *G-Suíte* é constituído de diferentes tipos de funções que envolvem restrições de igualdade linear, desigualdade linear e, também, desigualdade não linear. Todas as variantes descritas na Seção 3 foram testadas, juntamente com o APM e o método de tratamento de restrições proposto em [19]. Os testes foram feitos num total de 35 execuções independentes. Todos os testes foram realizados com 250000 avaliações e apenas os indivíduos factíveis foram considerados como solução, exceto na função G20, visto que essa função não possui solução factível conhecida na literatura.

Dado a complexidade de obtenção de soluções de alguns problemas, dois cenários de comparação foram definidos. No primeiro, considera-se os 11 primeiros problemas (G1 a G11), que possuem região de factibilidade mais ampla e, portanto, mais fácil de obter soluções candidatas factíveis. No segundo, considera-se todos os problemas.

A comparação dos resultados obtidos das funções *G-Suíte* foi realizada utilizando uma ferramenta gráfica chamada perfil de desempenho (*performance profiles*).

##### 4.1 Perfis de desempenho

Essa ferramenta foi proposta em [17] para facilitar a interpretação e visualização dos resultados obtidos em experimentos. Em geral, compara-se um conjunto de diferentes algoritmos de modo a se escolher aquele com o melhor desempenho sobre um certo conjunto de problemas.

Considere um conjunto  $P$  de problemas teste  $p_j$ , com  $j = 1, 2, \dots, n_p$ ,  $A$  um conjunto de algoritmos  $a_i$  com  $i = 1, 2, \dots, n_a$  e  $t_{p,a}$  uma métrica de desempenho (como, por exemplo, tempo computacional, média, etc.).

A razão de desempenho, que é sempre maior ou igual a 1, é definida como

$$r_{p,a} = \frac{t_{p,a}}{\min\{t_{p,a} : a \in A\}}. \quad (15)$$

O perfil de desempenho do algoritmo é dado por:

$$\rho_a(\tau) = \frac{1}{n_p} |\{p \in P : r_{p,a} \leq \tau\}|. \quad (16)$$

que é a fração de problemas resolvidos pelo algoritmo com desempenho dentro de um fator  $\tau$  do melhor desempenho obtido, considerando todos os algoritmos. O perfil é uma função constante por partes, não-decrescente e contínua.

Em adição às idéias propostas em [17], Barbosa et al. [18] sugere que a área sob a curva  $\rho_a$  é um indicador global de desempenho do algoritmo: quanto maior a área, maior a eficiência do algoritmo. Neste artigo usa-se este indicador de desempenho.

## 4.2 Primeiro cenário - G1 a G11

Para melhor visualização dos resultados, os métodos foram divididos em três conjuntos para gerar os perfis. O primeiro conjunto compara as variações do APM já propostas na literatura. Aos outros grupos, destinou-se a comparação entre as novas variantes para o APM e o método proposto em [19] que é referenciado aqui por DEB. Essas comparações podem ser observadas na Figura 1, que apresenta as áreas sob as curvas dos perfis de desempenho considerando os 11 primeiros problemas descritos em [16]. As comparações foram feitas dentro de cada conjunto levando em consideração as médias dos resultados encontrados por cada método de tratamento de restrições em cada problema, haja vista que todos os métodos testados tiveram quase que 100% de taxa de factibilidade (porcentagem de execuções em que se encontrou ao menos um indivíduo factível).

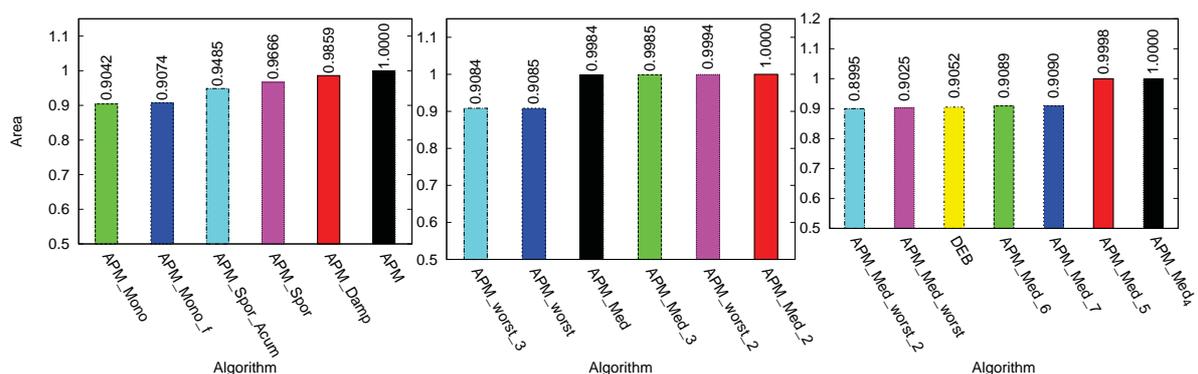


Figura 1: Áreas sob as curvas dos perfis de desempenho considerando os problemas G1 - G11.

É importante observar que o método **APM** se destaca na classificação dentre suas variações já propostas na literatura. Ele é seguido pelo método **APM\_Damp** e pelo **APM\_Spor**, nesta ordem.

Dentre as novas variantes, destacam-se, nessa classificação parcial, os métodos **APM\_Med\_2**, **APM\_Med\_4** e **APM\_Med\_5**. Na Figura 2 é apresentada a classificação geral com os melhores métodos obtidos:

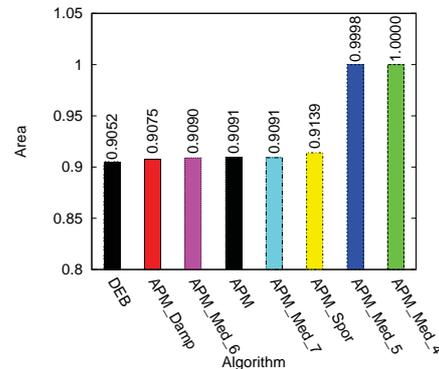


Figura 2: Classificação dos melhores métodos - G1 a G11.

Verifica-se que os métodos **APM\_Med\_4** e **APM\_Med\_5** obtiveram melhores desempenho em relações aos demais.

Percebe-se que, na seleção dos melhores métodos, o APM obteve melhor resultado do que o APM\_Spor. Já na classificação final, isso se inverte. Isso ocorre porque o APM\_Spor consegue resolver uma maior quantidade de problemas do que o APM. Em contrapartida, o APM consegue obter melhores soluções num maior número de problemas. Na primeira comparação, o valor máximo de  $\tau$  é menor do que o da segunda comparação. Enquanto no primeiro caso a área sob a curva do perfil de desempenho do APM é maior do que a do APM\_Spor, mesmo com a inversão da ordem dos perfis de desempenho. No segundo caso, essa troca de ordem dos perfis de desempenho, culmina numa área maior do que o APM\_Spor, pois a superioridade em relação a  $\rho$  é acumulada ao longo de um máximo valor de  $\tau$  maior que o anterior. Consegue-se perceber essa mudança na Figura 3, dado o fato de que os perfis desses métodos se invertem ao longo do eixo  $\tau$ .

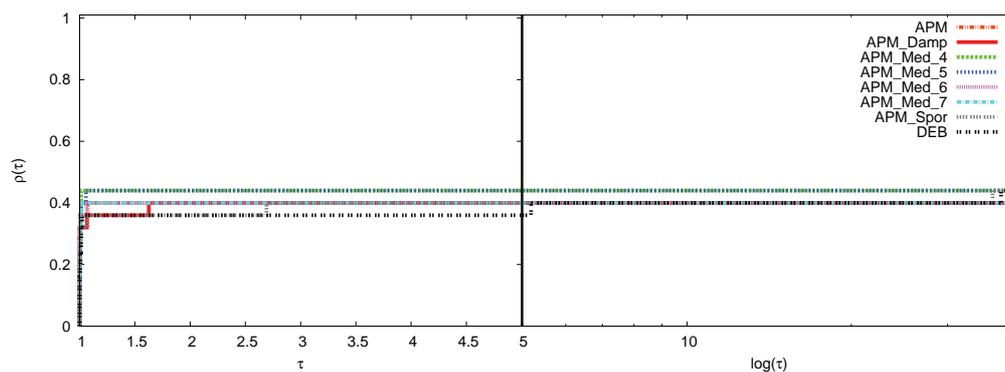


Figura 3: Áreas sob as curvas dos perfis de desempenho, considerando os melhores métodos segundo a Figura 1 e os problemas G1 - G11.

### 4.3 Segundo cenário - G1 a G24

A idéia deste cenário é encontrar o método mais robusto, já que a partir da função G11 há diferença significativa na taxa de factibilidade entre os métodos. A fim de comparar os métodos, a estratégia tomada foi a de considerar como critério de comparação o valor da média da função objetivo dividido pela taxa de factibilidade para cada método, indicado por [16].

Novamente os métodos foram divididos nos mesmos grupos e os melhores selecionados para uma comparação final. A Figura 4 mostra as comparações com os resultados parciais.

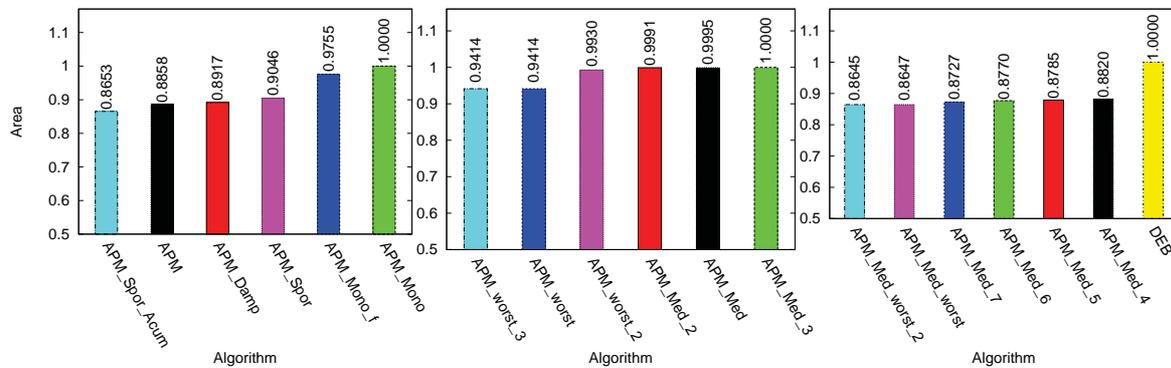


Figura 4: Áreas sob as curvas dos perfis de desempenho considerando os problemas G1 - G24.

Neste cenário, observa-se que o **APM\_Mono** é superior aos demais métodos já descritos em publicações passadas. Ele é seguido pelo **APM\_Mono.f**. Dentre os novos métodos, destacou-se o **APM\_Med\_3**, **APM\_Med\_4** e **APM\_Med\_5**. Estes métodos foram classificados para a próxima fase de comparação, que podem ser observados na Figura 5:

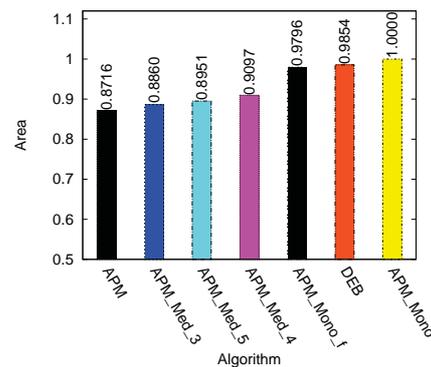


Figura 5: Áreas sob as curvas dos perfis de desempenho, considerando os melhores métodos segundo a Figura 4 e os problemas G1 - G24.

A Figura 6 apresenta os perfis de desempenho dos métodos para as funções de G1 a G24.

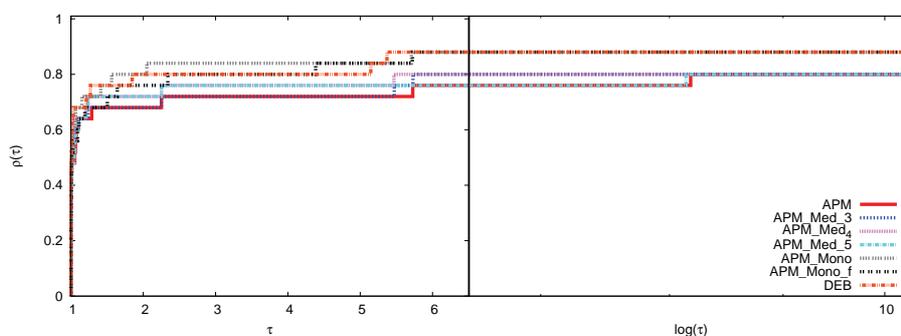


Figura 6: Perfis de desempenho dos métodos de G1 a G24

A variante **APM\_Mono** obteve maior sucesso, seguido do método **DEB** e **APM\_Mono.f**. Dentre os novos métodos, **APM\_Med\_4** e **APM\_Med\_5** foram os melhores classificados.

## 5. Conclusão

No primeiro cenário, dentro da perspectiva de problemas menos complexos, os métodos **APM\_Med\_4** e **APM\_Med\_5** foram mais bem sucedidos tanto em relação aos outros novos métodos quanto aos métodos já existentes. Há indícios de que os métodos **APM\_Med\_4** e **APM\_Med\_5** apresentam melhor ajuste nesse cenário observando a Figura 2 e a Figura 5.

No segundo cenário, que privilegia a robustez, visualiza-se a competitividade dos métodos **APM\_Mono**, **DEB** e **APM\_Mono.f**. Estes métodos fazem parte do conjunto de métodos já conhecidos na literatura. Devemos observar que, apesar do método **APM\_Mono** já ter sido proposto anteriormente, ele ainda não tinha sido testado antes da maneira proposta nesse trabalho. Conclui-se que as variantes *Monotonic* agregam robustez ao **APM**, já que ocuparam as primeiras posições nos desempenhos mostrados na Figura 5. Os métodos **APM\_Med\_4** e **APM\_Med\_5** foram os próximos melhores classificados.

Na perspectiva desses dois cenários e considerando a proposta desse trabalho, pode-se concluir que os métodos **APM\_Med\_4** e **APM\_Med\_5** são os mais competitivos. Vemos que eles se comportam muito bem nos problemas em que suas regiões de factibilidade são mais amplas ou de fácil obtenção de soluções. Quando inseridos em problemas de complexidade maior, eles apresentam dificuldades por não encontrar soluções candidatas factíveis em algumas das execuções.

Dentre os métodos já propostos na literatura e descritos nesse trabalho, destaca-se o método **DEB**, que além de ser de fácil implementação, apresenta bom desempenho na busca de soluções factíveis, mesmo nos problemas em que a região de factibilidade é complexa.

Na comparação do método **APM** e suas variantes já propostas anteriormente, destaca-se os métodos **APM\_Mono** e **APM\_Mono.f** que apresentaram robustez na solução dos problemas mais complexos.

O perfil de desempenho demonstrou ser uma ferramenta bastante interessante para interpretação dos resultados obtidos neste artigo assim como aqueles analisados em [18].

## 6. Agradecimentos

Os autores agradecem às agências de fomento pelos seguintes apoios CNPq (306815/2011-7 e 300192/2012-6) e FAPEMIG (grant no. TEC PPM 528/11).

## Referências

- [1] **Shoenauer, M. and Michalewicz, Z.**, Evolutionary computation at the edge of feasibility. In H-M. Voigt; W. Ebeling; I. Rechenberg and H-P. Schwefel, editors, Parallel Problem Solving from Nature - PPSN IV, volume 1141, pages 245-254, Berlin, 1996. Springer - Verlag. LNCS.
- [2] **Koziel, S. and Michalewicz., Z.**, Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. *Evolutionary Computation*, 7(1):19-44, 1999.
- [3] **Liepins, G.E. and Potter, W.D.**, A Genetic Algorithm Approach to Multiple-Fault Diagnosis. In Lawrence Davis, editor, *Handbook of Genetic Algorithms*, chapter 17, pages 237-250. Van Nostrand Reinhold, New York, New York, 1991.
- [4] **Orvosh, D. and Davis, L.**, Using a Genetic Algorithm to Optimize Problems with Feasibility Constraints. In *Proceedings of the First IEEE Conference on Evolutionary*

- Computation, pages 548-553. IEEE Press, 1994.
- [5 ] **Adeli, H. and Cheng, N.-T.**, Augmented Lagrangian Genetic Algorithm for Structural Optimization. *Journal of Aerospace Engineering*, 7(1):104-118, January 1994.
  - [6 ] **Barbosa, H.J.C.**, A coevolutionary genetic algorithm for constrained optimization problems. In *Proc. of the Congress on Evolutionary Computation*, pages 1605-1611, Washington, DC, USA, 1999.
  - [7 ] **Surry, P.D. and Radcliffe, N.J.**, The COMOGA Method: Constrained Optimisation by Multiobjective Genetic Algorithms. *Control and Cybernetics*, 26(3), 1997.
  - [8 ] **Runarsson, T.P. and Yao, X.**, Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 4(3):284-294, 2000.
  - [9 ] **Kampen, A.H.C. van, Strom, C.S. and Buydens, L.M.C.**, Lethalization, penalty and repair functions for constraint handling in the genetic algorithm methodology. *Chemometrics and Intelligent Laboratory Systems*, 34:55-68, 1996.
  - [10 ] **Michalewicz, Z. and Shoenauer, M.**, Evolutionary algorithms for constrained parameter optimization problems. *Evolutionary Computation*, 4(1):1-32, 1996.
  - [11 ] **Hinterding, R. and Michalewicz, Z.**, Your brains and my beauty: Parent matching for constrained optimization. In *Proc. of the Fifth Int. Conf. on Evolutionary Computation*, pages 810-815, Alaska, May 4-9 1998.
  - [12 ] **Koziel, S. and Michalewicz, Z.**, A decoder-based evolutionary algorithm for constrained optimization problems. In T. Bäck; A.E. Eiben; M. Shoenauer and H.-P. Schwefel, editors, *Proc. of the Fifth Parallel Problem Solving from Nature*, Amsterdam, September 27-30 1998. Springer Verlag. *Lecture Notes in Computer Science*.
  - [13 ] **Kim, J.-H. and Myung, H.**, Evolutionary programming techniques for constrained optimization problems. *IEEE Trans on Evolutionary Computation*, 2(1):129-140, 1997.
  - [14 ] **Barbosa, H. J. C. and Lemonge, A. C. C.**, An adaptive penalty scheme in genetic algorithms for constrained optimization problems. In *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 287-294, New York, 9-13 July 2002. Morgan Kaufmann Publishers.
  - [15 ] **Barbosa, H. J. C. and Lemonge, A. C. C.**, An adaptive penalty method for genetic algorithms in constrained optimization problems. In Hitoshi Iba, editor, *Frontiers in Evolutionary Algorithms*, pages 9-34. I-Tech, 2008.
  - [16 ] **Liang, J. J., et al.**, Problem definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization. Tech Report, 2006.
  - [17 ] **Dolan, E. D., More, J.**, Benchmarking optimization software with performance profiles, *Mathematical Programming*, 2002; v. 91, 201:213.
  - [18 ] **Barbosa, H. J. C., Bernardino, H. S. e Barreto, A. M. S.**, Using Performance Profiles to Analyze the Results of the 2006 CEC Constrained Optimization Competition. In: *2010 IEEE World Congress on Computational Intelligence - WCCI, 2010, Barcelona*.
  - [19 ] **Deb, K.**, An Efficient Constraint Handling Method for Genetic Algorithms, *Comp. Meth. Appl. Mech. Eng.*, 2000; 311:338.