

Uso da Meta-Heurística otimização por
exame de partículas no planejamento FlorestalUse of Metaheuristics particle swarm
optimization in Forest planningFlavio Augusto Ferreira do Nascimento¹, Andrea Nogueira Dias²,
Afonso Figueiredo Filho³, Julio Eduardo Arce⁴ e Gabriel de Magalhães Miranda⁵

Resumo

O objetivo do trabalho foi testar a aplicabilidade no planejamento florestal de duas abordagens do algoritmo PSO (momento de inércia e fator de restrição) combinadas com duas topologias de vizinhança (*gbest* e *lbest*). O problema de planejamento florestal foi formulado de acordo com o modelo tipo I de Johnson e Scheurmann (1977) com o objetivo de maximizar o retorno financeiro, considerando-se como restrição, a regulação do fluxo de produção em um horizonte de planejamento de 26 anos. Foram geradas 2.646 variáveis de decisão para o problema. A avaliação do algoritmo PSO foi realizada com base na média, desvio padrão, coeficiente de variação e valor máximo e mínimo das soluções (*fitness*) de 30 rodadas independentes de cada abordagem testada. A fim de definir as diferenças entre as abordagens/topologias foram utilizados os testes não paramétricos de Kruskal-Wallis e Dunn ($\alpha = 0,05$). Os resultados obtidos indicam que abordagens/topologias apresentaram diferenças quanto à qualidade das soluções obtidas. A melhor eficácia foi obtida com a abordagem momento de inércia utilizando a topologia *gbest* (PSOw *gbest*) a qual apresentou valor de 99,07% em relação ao ótimo matemático. Considerando-se as demais estatísticas de comparação e a capacidade de convergência, melhores resultados foram obtidos com abordagem/topologia fator de restrição combinada à topologia *gbest* (PSOx *gbest*).

Palavras-chave: Programação linear inteira, modelo tipo I, análise combinatória.

Abstract

The objective of this work was to test two approaches to the PSO algorithm (inertia weight and constriction factor) combined with two neighborhood topologies (*gbest* and *lbest*) and determine the best of these. The test problem of forest planning was formulated according to Johnson and Scheurmann (1977) type I model in order to maximize financial return. 2.646 decision variables were generated. Variables mean, standard deviation, coefficient of variation, maximum and minimum on the value of the objective function (*fitness*) were measured on 30 replications. In order to define the differences between the approaches/topologies, the nonparametric Kruskal-Wallis and Dunn tests were employed at 5% probability. According to the results, approaches/topologies differed in the quality of solutions. The best efficacy was obtained with the inertia weight approach with the *gbest* topology (PSOw *gbest*) which gave a value of 99.07%. Considering the other statistics comparing and the convergence ability, better approach/topology of PSO was the constriction factor combined with the *gbest* topology (PSOx *gbest*).

Keywords: Integer linear programming, model type I, combinatorial analysis.

¹Engenheiro Florestal, Doutorando em Engenharia Florestal. UFPR - Universidade Federal do Paraná/Departamento de Engenharia Florestal - Rua Lothário Meissner, 900, Jardim Botânico 80210-170 Curitiba, PR. E-mail: fafnascimento@hotmail.com

²Engenheira Florestal, Doutora. UNICENTRO - Universidade Estadual do Centro-Oeste/ Departamento de Engenharia Florestal - Rodovia PR 153, Km 7 - Riozinho Caixa postal 21 - 84.500.000. E-mail: andias@irati.unicentro.br

³Engenheiro Florestal, Prof. Sênior do Curso de Pós-Graduação em Engenharia Florestal. UFPR - Universidade Federal do Paraná/ Departamento de Engenharia Florestal - Rua Lothário Meissner, 900, Jardim Botânico 80210-170 Curitiba, PR - Pesquisador do CNPq 1C. e-mail: afonso.figueiredo@pq.cnpq.br

⁴Julio Eduardo Arce, Engenheiro Florestal, Professor Dr. do Departamento de Ciências Florestais. UFPR - Universidade Federal do Paraná - Rua Lothário Meissner, 900, Jardim Botânico 80210-170 Curitiba, PR. E-mail: jarce@ufpr.br

⁵Gabriel de Magalhães Miranda, Engenheiro Florestal, Doutor. do UNICENTRO - Universidade Estadual do Centro-Oeste/ Departamento de Engenharia Florestal - Rodovia PR 153, Km 7 - Riozinho Caixa postal 21 - 84.500.000. E-mail: gmiranda@irati.unicentro.br

INTRODUÇÃO

O planejamento florestal, devido à sua natureza, tem como característica principal, a resolução de problemas envolvendo elevado número de variáveis, longos horizontes de planejamento, variações de clima e mudanças de mercado, tais como flutuações de preços, de oferta e de demanda. O uso de técnicas que garantam tomadas de decisões que sejam as melhores possíveis é essencial para a adequada alocação dos recursos visando à sustentabilidade do empreendimento (NASCIMENTO, 2010).

Nas questões ligadas à regulação florestal, é comum a busca das melhores alternativas de manejo para a floresta de forma a maximizar o retorno financeiro ou minimizar os custos de produção respeitando restrições como, por exemplo, manter constante a produção de madeira durante os períodos do horizonte de planejamento.

Os problemas de planejamento florestal são normalmente formulados via programação linear (PL) ou, quando é exigido que as variáveis de decisão assumam apenas valores inteiros, são formulados via programação linear inteira (PLI). No caso dos problemas de PLI com grande número de variáveis de decisão, o tempo computacional dos métodos exatos de otimização é elevado, o que os torna, muitas vezes, tecnicamente inviáveis.

Como alternativa para resolução de problemas de PLI, técnicas denominadas meta-heurísticas podem ser utilizadas. De acordo com Goldberg e Luna (2005), são técnicas que buscam alcançar boas soluções utilizando um esforço computacional razoável, sendo capazes de garantir a viabilidade e, em alguns casos, a otimalidade da solução encontrada.

Algumas destas técnicas meta-heurísticas são aplicadas no planejamento florestal, cabendo citar: algoritmo genético (DUCHEYNE *et al.*, 2004; GOMIDE *et al.*, 2009 e RODRIGUES *et al.*, 2004a), algoritmo busca tabu (BETTINGER *et al.*, 2007 e RODRIGUES *et al.*, 2003), *simulated annealing* (PEREIRA, 2004 e RODRIGUES *et al.*, 2004b) e a otimização por enxame de partículas (PUKKALA; KURTILA, 2009).

O algoritmo de otimização por enxame de partículas (*particle swarm optimization*, PSO) é uma técnica que tem se destacado pela sua simplicidade, robustez e eficiência. O seu desenvolvimento se baseia no comportamento coletivo de animais que vivem em sociedade, tais como enxame de abelhas, bando de pássaros e cardu-

me de peixes. Segundo Mendes (2004), a PSO apresenta as seguintes vantagens: (1) simplicidade de implementação; (2) existência de poucos parâmetros a serem ajustados; (3) utilização de uma população relativamente pequena; e (4) necessidade de um número relativamente pequeno de avaliações da função objetivo para convergir.

O algoritmo PSO foi apresentado em 1995 (KENNEDY; EBERHART, 1995) e desde então, várias modificações foram sugeridas ao método, dentre as quais, a abordagem que utiliza o momento de inércia (SHI; EBERHART, 1998) e a abordagem com o fator de restrição (CLERC; KENNEDY, 2002). Ambas modificam a forma como a velocidade das partículas é atualizada a cada iteração do algoritmo.

Outras variações do algoritmo são as diferentes topologias de vizinhança que podem ser empregadas. Estas definem a forma como as partículas interagem durante a execução do algoritmo. Dentre as mais utilizadas, têm-se as topologias denominadas *gbest* (*global best*), em que todas as partículas estão conectadas às demais e *lbest* (*local best*), em que cada partícula conecta-se a apenas *k* partículas, sendo comum empregar o valor de *k* igual a 2. Mendes (2004) e Guo *et al.* (2006) descreveram detalhadamente esse assunto.

O objetivo deste trabalho foi testar duas abordagens do algoritmo PSO (momento de inércia e fator de restrição) combinadas a dois tipos de topologia de vizinhança (*gbest* e *lbest*) a fim de verificar qual abordagem/topologia é a mais adequada na resolução do modelo tipo I de planejamento florestal.

MATERIAL E MÉTODOS

Área de estudo

O estudo utilizou a base de dados do ano de 2007 de uma empresa florestal da região norte do Estado de Santa Catarina. As áreas de plantio compreendem 96% de *Pinus taeda* e o restante de *Pinus elliottii*. Os povoamentos tinham, na época do estudo, idades variando de 1 a 33 anos. Adotava-se como manejo padrão desbastes aos 10 anos e corte final aos 17 anos, sendo a produção destinada totalmente a produtos de madeira sólida.

Grande parte dos povoamentos, cerca de 70%, tem idade entre 1 e 10 anos e, ainda, 26% são povoamentos antigos com mais de 20 anos de idade. Esta condição aumenta a complexidade do planejamento florestal, pois se torna difícil encontrar uma solução que atenda as restrições de demanda durante todo o horizonte de planejamento.

Descrição do problema de planejamento florestal

O problema de planejamento florestal foi formulado utilizando o modelo clássico tipo I, conforme definido por Johnson e Scheurmann (1977). Tendo como função objetivo a maximização do valor periódico equivalente (VPE) do empreendimento. Considerou-se como restrição, a regulação do fluxo de produção. O VPE foi calculado de acordo com a metodologia apresentada em Silva (2005).

Os limites de produção volumétrica foram fixados em 2.000 m³ (mínimo) e 22.000 m³ (máximo) para cada período do horizonte de planejamento (HP), que neste caso foi de 26 anos, seguindo a orientação de Clutter *et al.* (1983), os quais sugeriram o emprego de um HP de pelo menos uma vez e meia da rotação dos povoamentos.

Foram consideradas como possíveis alternativas de manejo idades próximas ao manejo padrão da empresa, cujos desbastes podem ser realizados entre 9 e 11 anos e o corte raso entre 16 e 21 anos de idade. Foram geradas 2.646 variáveis de decisão, sendo que para possibilitar a resolução do problema por algoritmos exatos foram empregadas apenas 30 unidades de gestão (talhões) do total de 628 da empresa.

A formulação matemática do problema, conforme Johnson e Scheurmann (1977), é apresentada a seguir:

$$\text{Maximizar } VPE_G = \sum_{i=1}^M \sum_{j=1}^N C_{ij} \cdot X_{ij}$$

sujeito a:

$$\sum_{j=1}^N X_{ij} = 1 \quad \forall i=1,2,\dots,M \quad X_{ij} \in \{0,1\} \text{ em que:}$$

$$X_{ij} = \begin{cases} 1, & \text{se a prescrição } j \text{ for adotada na unidade de gestão } i \\ 0, & \text{caso contrário} \end{cases}$$

$$\sum_{i=1}^M \sum_{j=1}^N VT_{ijk} \cdot X_{ij} \geq Pmin_k \quad (k = 1,2,3, \dots, 26)$$

$$\sum_{i=1}^M \sum_{j=1}^N VT_{ijk} \cdot X_{ij} \leq Pmax_k \quad (k = 1,2,3, \dots, 26)$$

em que:

VPE_G = valor periódico equivalente global da floresta;

C_{ij} = valor periódico equivalente de cada talhão i, manejado sob a prescrição j;

X_{ij} = talhão i assinalado à prescrição j;

M = número de talhões;

N = número total de alternativas de manejo j no talhão i;

VT_{ijk} = volume total (m³) produzido pelo i-ésimo talhão assinalado na j-ésima alternativa de manejo, no início do período k;

Pmin_k = produção mínima estabelecida para o período k, em m³;

Pmax_k = produção máxima estabelecida para o período k, em m³.

Algoritmo Otimização por Enxame de Partículas

A PSO é uma técnica que se baseia no movimento coletivo de um grupo de partículas: o enxame de partículas. Cada membro deste enxame é movimentado através do espaço de busca do problema por duas forças. Uma os atrai com uma magnitude aleatória para a melhor localização já encontrada por ele próprio (*pbest*) e outra para a melhor localização encontrada entre alguns ou todos os membros do enxame (*gbest*). A posição e a velocidade de cada partícula são atualizadas a cada iteração até todo o enxame convergir (CASTRO, 2007).

O algoritmo (Figura 1) inicia cada partícula com valores aleatórios de posição e velocidade. Durante a sua execução cada partícula avaliará sua posição atual (ou solução atual) em relação à melhor posição já encontrada por ela mesma, fazendo com que o valor de *pbest* seja atualizado. Cada partícula também avaliará a qualidade da melhor solução encontrada na sua vizinhança, permitindo a atualização do valor de *gbest*. A avaliação da qualidade de uma posição (ou solução) é realizada por meio da função de fitness ou aptidão. Após atualizar o valor de velocidade com os novos valores de *pbest* e *gbest*, cada partícula irá se deslocar para uma nova posição.

O algoritmo PSO não trabalha diretamente com restrições. Uma estratégia para que este algoritmo manipule restrições, é utilizando funções de penalidade. Assim, a função objetivo penalizada *f_p(x)* é obtida através da modificação da função objetivo *f(x)*, da seguinte forma:

$$f_p(x) = f(x) - vp \cdot VT$$

em que:

f_p(x) = valor da função-objetivo penalizada para a solução x;

f(x) = valor da função-objetivo do problema;

vp = penalização (R\$/m³) para cada unidade de produção violada;

VT = violação total (m³) das restrições de produção (mínima e máxima).

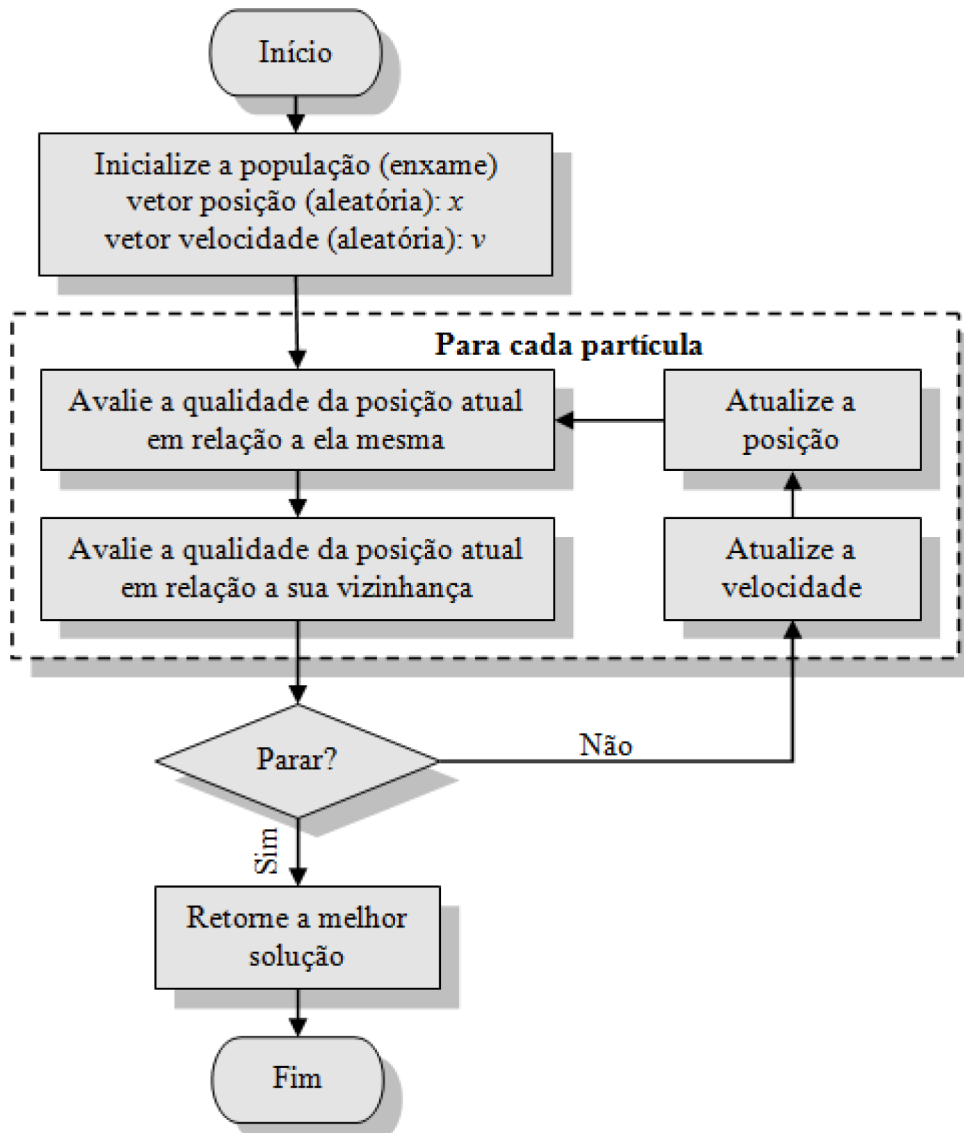


Figura 1. Fluxograma de funcionamento do algoritmo PSO.
Figure 1. Flowchart of the PSO algorithm operation.

A penalização (v_p) foi configurada em R\$/m³ 100.000,00 para garantir que toda solução que viole as restrições apresente valor negativo para a função-objetivo penalizada e seja considerada inviável.

Foram avaliadas duas abordagens para a atualização da velocidade das partículas a cada iteração: momento de inércia e fator de constrição.

De acordo com Shi e Eberhart (1998), na abordagem com momento de inércia (PSOw) a equação da velocidade assume a seguinte forma:

$$v_p^{(it+1)} = w \cdot v_p^{it} + c_1 \cdot rand_1^{(it)} \cdot (pbest_p^{(it)} - x_p^{(it)}) + c_2 \cdot rand_2^{(it)} \cdot (gbest_p^{(it)} - x_p^{(it)})$$

em que:

- v_p = velocidade da partícula p;
- c_1 e c_2 = coeficientes cognitivos e social;
- $rand$ = função aleatória de distribuição uniforme entre 0 e 1;
- $pbest_p$ = melhor posição que a partícula p já obteve durante a busca;

$gbest_p$ = melhor posição encontrada na vizinhança da partícula p;
 it = iteração atual; e
 w = momento de inércia.

A atualização do valor do momento de inércia a cada iteração foi feita conforme a fórmula proposta por Eberhart e Shi (2000).

Segundo Clerc e Kennedy (2002), na abordagem com fator de constrição (PSOx) a equação de velocidade é escrita da seguinte forma:

$$v_p^{(it+1)} = \chi \cdot [v_p^{it} + c_1 \cdot rand_1^{(it)} \cdot (pbest_p^{(it)} - x_p^{(it)}) + c_2 \cdot rand_2^{(it)} \cdot (gbest_p^{(it)} - x_p^{(it)})]$$

em que χ é o coeficiente de constrição e é calculado com a equação:

$$\chi = \frac{2 \cdot k}{|2 - \varphi - \sqrt{\varphi^2 - 4 \cdot \varphi}|}$$

onde: $\varphi = c_1 + c_2, \varphi > 4$, com φ na maioria das vezes igual a 4,1 (tendo assim $c_1 = c_2 = 2,05$) e $k = 1$, o que determina um $\chi \approx 0,729$.

A atualização da posição de cada partícula em cada iteração foi realizada da mesma forma para as duas abordagens, com a seguinte equação:

$$x_p^{(it+1)} = x_p^{(it)} + v_p^{(it+1)}$$

em que:

x_p = posição da partícula p .

Com base na revisão de literatura e teste iniciais do algoritmo PSO definiram-se os parâmetros de configuração para as abordagens testadas. Estes são apresentados na Tabela 1.

Tabela 1. Parâmetros de configuração das abordagens testadas do algoritmo PSO.

Table 1. Configurations parameters of the tested approach by PSO algorithm.

Parâmetro	Abordagem da PSO	
	Momento de inércia (PSOw)	Fator de construção (PSOx)
População	50	50
Coefficiente cognitivo (c_1)	2	2,05
Coefficiente social (c_2)	2	2,05
Inércia inicial (w_{ini})	0,9	-
Inércia final (w_{fin})	0,4	-
V_{max}	10%	100%
Iterações	3.000	3.000

Também foram avaliadas para cada abordagem do algoritmo PSO duas topologias de vizinhança: *gbest* e *lbest* (Figura 2).

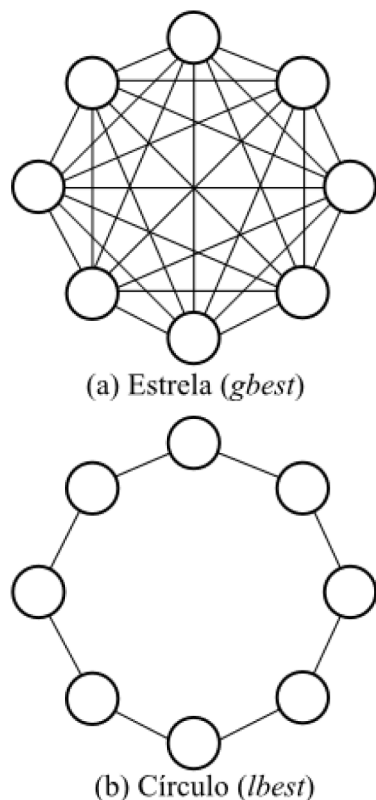


Figura 2. Topologias de vizinhança utilizadas.
Figure 2. Neighborhood topologies used.

Implementação computacional e análise estatística

As abordagens testadas do algoritmo PSO foram implementadas em computador utilizando a linguagem C++. Foi empregado o ambiente de desenvolvimento Dev-C++, um software livre distribuído sob os termos da GNU General Public License. Os algoritmos foram executados em um computador com processador AMD Sempron 1,60 GHz com 512 MB de memória RAM, tanto para o algoritmo *branch-and-bound* (B&B) quanto para as abordagens/topologias da PSO. O algoritmo B&B foi empregado para obtenção da solução ótima, esta foi utilizada como valor de referência na avaliação das soluções da PSO.

A avaliação do algoritmo PSO foi realizada com base na média, desvio padrão, coeficiente de variação e valor máximo e mínimo das soluções (*fitness*) de 30 experimentos independentes de cada abordagem testada. Este valor foi escolhido devido ao número de variáveis do problema, o qual exige considerável tempo de processamento. Mendes (2004) sugere que ao mínimo devem ser realizados 30 experimentos.

A amostra foi então submetida ao teste não paramétrico de Kruskal-Wallis ($\alpha = 0,05$), também utilizado por Gomide *et al.* (2009), uma vez que não foi possível atender as pressuposições para utilização da ANOVA, como homogeneidade das variâncias e normalidade dos dados. Para distinguir os grupos foi utilizado o teste de Dunn ($\alpha = 0,05$), um procedimento não-paramétrico de comparação múltiplas semelhante ao teste de Tukey.

O teste de Kruskal-Wallis foi processado no R, software livre distribuído sob os termos da GNU *General Public License* versão 2 (R DEVELOPMENT CORE TEAM, 2006). Já o teste de Dunn foi calculado conforme a metodologia apresentada em Callegari-Jacques (2003).

O algoritmo PSO também foi avaliado em relação ao algoritmo exato de resolução de problemas de PLI, *branch-and-bound*, implementado no software LINGO versão 9.0. Para tanto, foram utilizadas duas medidas, a eficácia e a eficiência. Por meio da eficácia foi possível avaliar a qualidade da solução encontrada pelo algoritmo PSO em relação ao ótimo matemático, obtido com o algoritmo exato *branch-and-bound*. A eficiência, que mede o quão rápido é o algoritmo, foi analisada pela comparação dos tempos de processamento dos algoritmos PSO e *branch-and-bound*, utilizando para ambos os mesmos recursos computacionais. As duas medidas fo-

ram calculadas de acordo com a metodologia apresentada em Rodrigues *et al.* (2003).

RESULTADOS E DISCUSSÕES

A solução ótima do problema para o VPE, encontrada por meio do algoritmo exato *branch-and-bound*, foi de R\$ 516.474,20, sendo que a mesma requereu 1.064 segundos para ser alcançada. A eficácia do algoritmo PSO é apresentada na Tabela 2, para cada uma das abordagens/topologias testadas.

A melhor solução foi encontrada com a PSOW *gbest* com 99,07% do ótimo matemático, no entanto, esta apresentou as piores soluções em relação a média e o valor mínimo. A eficácia obtida com o algoritmo PSO ficou próxima da encontrada por outros pesquisadores. Como exemplo, Gomide *et al.* (2009) obtiveram 99,09% com o algoritmo genético, Pereira (2004) conseguiu 99,69% com o algoritmo *simulated annealing* e Rodrigues *et al.* (2003) alcançaram 98,84% com o algoritmo busca tabu.

As abordagens/topologias da PSO avaliadas foram cerca de 20 vezes mais rápidas que o algoritmo *branch-and-bound*. O fato de o método exato ter apresentado tempo computacional viável se deve ao reduzido número de variáveis de decisão do problema testado. A prática tem demonstrado que na medida em que se aumenta o tamanho do problema o requerimento computacional dos algoritmos exatos cresce exponencialmente. Isto por se tratar de uma classe de problemas denominados *NP-Hard*, os quais não possuem um algoritmo exato ca-

paz de obter a solução ótima em tempo computacional razoável.

As estatísticas de comparação calculadas com o VPE (Função Objetivo) de 30 rodadas das abordagens/topologias avaliadas da PSO são apresentadas na Tabela 3.

As abordagens momento de inércia (PSOW) e fator de restrição (PSOX) apresentaram os dois maiores coeficientes de variação quando combinadas com a topologia *gbest*. Nesta topologia cada partícula está conectada a todas as outras do enxame, o que faz com que elas se agrupem rapidamente em uma mesma região do espaço de busca. Isto, em muitos casos, pode fazer o algoritmo convergir prematuramente para ótimos locais. No entanto, a aglomeração de partículas permite ao algoritmo refinar a busca local, o que, em algumas vezes, garante encontrar melhores soluções, no caso, esta topologia apresentou as duas soluções com maiores VPE.

Na Figura 3 são apresentados os pontos para a eficácia e o boxplot, o qual ilustra os valores da mediana, percentis 25 e 75%, máximo e mínimo e ainda, os valores *outliers* das 30 soluções (*fitness*) obtidas no experimento.

Como pode ser visto na Figura 3, a PSOW *gbest*, apesar de obter a melhor solução das abordagens avaliadas, apresentou o menor valor para a mediana com VPE de R\$ 490.044,06, ou seja, 50% das soluções estão abaixo deste valor. Por outro lado, a PSOX *gbest* foi melhor para o problema formulado, pois apresentou a melhor mediana, com valor de VPE igual a R\$ 499.286,30, melhor valor mínimo e percentil 75%.

Tabela 2. Eficácia das abordagens/topologias testadas em relação ao ótimo matemático e tempo médio de solução.
Table 2. Effectiveness of approaches/topologies tested in relation to the mathematical optimum and mean solution time.

Abordagem/ topologia	Eficácia (%)			Tempo médio de solução (segundos)
	Máxima	Média	Mínima	
PSOW <i>gbest</i>	99,07	95,29	92,73	52
PSOW <i>lbest</i>	98,43	96,27	94,74	53
PSOX <i>gbest</i>	98,56	96,59	94,56	52
PSOX <i>lbest</i>	98,04	96,16	94,14	52
<i>Branch-and-bound</i>	-	-	-	1064

Tabela 3. Média, desvio padrão, coeficiente de variação, valores máximos e mínimos do VPE das abordagens/topologias avaliadas.

Table 3. Mean, standard deviation, coefficient of variation, maximum and minimum values of VPE approach/topologies evaluated.

Abordagem	Média (R\$)	Desvio padrão (R\$)	CV (%)	Máximo (R\$)	Mínimo (R\$)
PSOW <i>gbest</i>	492.167,96	8.776,97	1,7833	511.668,47	478.911,84
PSOW <i>lbest</i>	497.225,75	4.815,11	0,9684	508.345,31	489.332,50
PSOX <i>gbest</i>	498.879,61	5.987,16	1,2001	509.036,75	488.358,94
PSOX <i>lbest</i>	496.641,37	4.630,31	0,9323	506.353,59	486.229,47

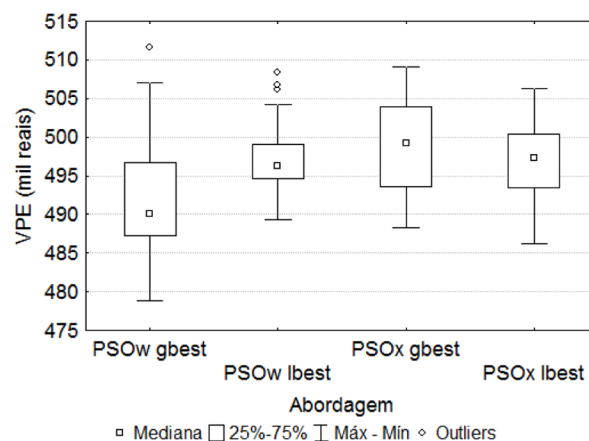
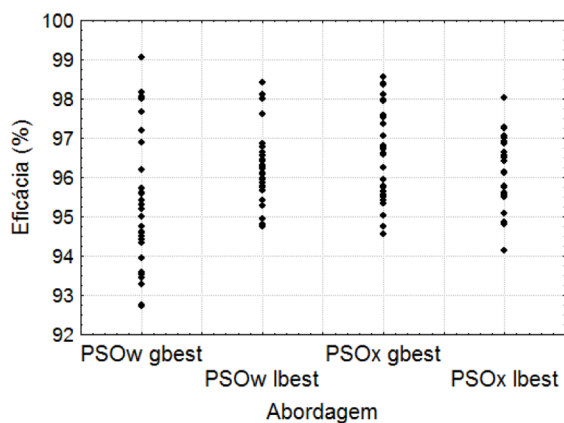


Figura 3. Comparação das soluções das abordagens/topologias testadas.
Figure 3. Comparison of the solution approach/topologies tested.

O desenvolvimento do VPE ao longo das iterações dos algoritmos das abordagens/topologias avaliadas é apresentado na Figura 4. Optou-se por apresentar a partir da iteração de número 100, uma vez que, a partir desta iteração, todas as abordagens/topologias já apresentavam soluções factíveis.

Observa-se na evolução da melhor solução da PSOx gbest que esta possui grande capacidade de convergência, sendo que foram necessárias apenas 808 iterações para se encontrar a melhor solução. De acordo com os resultados reportados por Eberhart e Shi (2000) esta abordagem é em média mais rápida para convergir que a PSO com momento de inércia (PSOW), porém tem a tendência de ficar estagnada em ótimos locais.

Por meio da prova não paramétrica de Kruskal-Wallis, rejeitou-se a hipótese H_0 (p -valor $< 0,05$), ou seja, ao menos uma das abordagens/topologias é diferente das demais em relação às soluções (fitness) obtidas. Os resultados do teste de Dunn, para avaliar quais grupos diferiram entre si, são apresentados na Tabela 4.

Tabela 4. Resultados do teste de Dunn para o problema 1.

Table 4. Dunn's tests results for the problem 1.

Algoritmos	Soma dos postos	Postos médios	Comparação dos grupos*
PSOx gbest	2192	73,07	a*
PSOW lbest	1965	65,50	a
PSOx lbest	1902	63,40	a b
PSOW gbest	1201	40,03	b

*Abordagens seguidas de mesma letra não diferem estatisticamente pelo teste de Dunn ($\alpha = 0,05$)

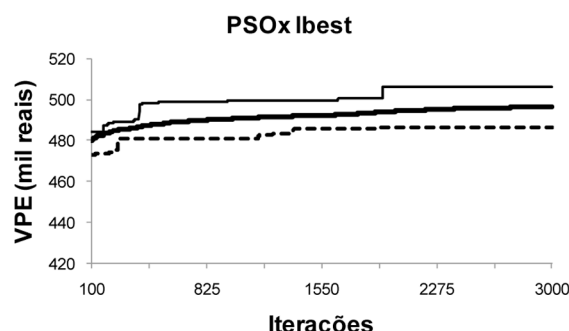
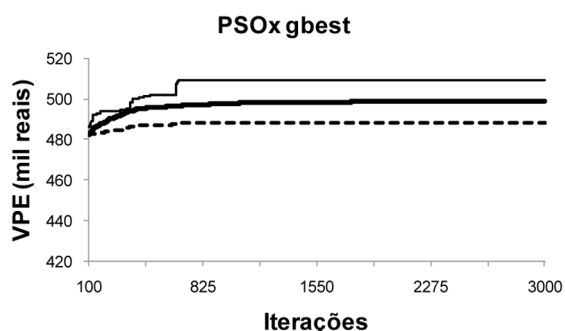
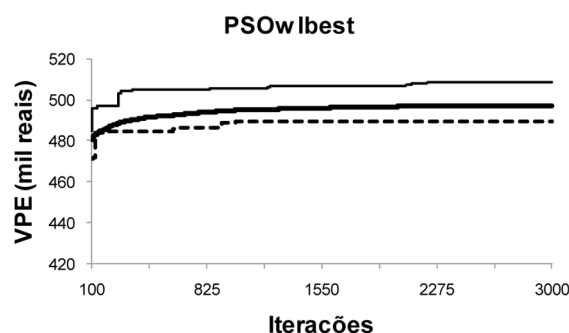
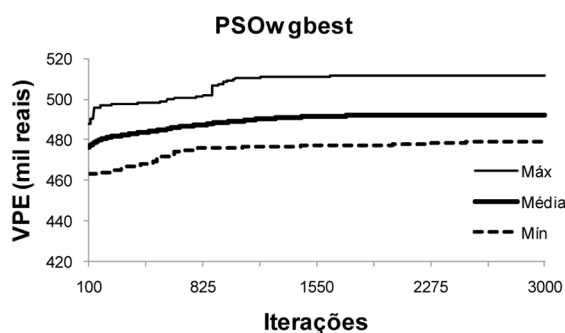


Figura 4. Desenvolvimento do VPE da média, piores e melhores soluções nas iterações dos algoritmos da PSO.
Figure 4. Development mean VPE, worst and best solutions in the iterations of the PSO algorithms.

De acordo com o teste de Dunn, as melhores abordagens/topologias são a PSOx *gbest* e PSOw *lbest*. Para a PSOx *lbest* os resultados são inconclusivos, uma vez que a mesma não é estatisticamente diferente a PSOw *gbest*. Apesar de apresentar a melhor solução para o problema, a PSOw *gbest* ficou em último lugar de acordo com o teste de Dunn. Isto se deve ao fato do teste avaliar toda a distribuição de soluções encontrada pelo algoritmo, que para esta abordagem/topologia mostrou-se pior que as demais.

CONCLUSÕES

- As abordagens do algoritmo PSO avaliadas na presente pesquisa apresentaram diferenças quanto à eficácia, não diferindo, porém, quanto à eficiência.
- A PSOx *gbest* é a abordagem mais indicada para o problema de planejamento florestal com variáveis inteiras, por ter apresentado os melhores valores de média e mediana, possuir maior capacidade de convergência e menor número de parâmetros a serem ajustados.
- O algoritmo PSO pode ser aplicado na resolução de problemas de planejamento florestal com variáveis inteiras, tendo desempenho semelhante ao de outras meta-heurísticas encontradas na literatura.

REFERÊNCIAS BIBLIOGRÁFICAS

BETTINGER, P.; BOSTON, K.; KIM, Y.; ZHU, J. Landscape-level optimization using tabu search and stand density-related forest management prescriptions. *European Journal of Operations Research*, Amsterdam, v.176, n.2, p.1265-1282. 2007.

CALLEGARI-JACQUES, S.M. **Bioestatística: princípios e aplicações**. Porto Alegre: Artmed, 2003. 264p.

CASTRO, E.G.; TSUZUKI, M.S.G. Simulation optimization using swarm intelligence as tool for cooperation strategy design in 3d predator-prey game. In: CHAN, F.T.S.; TIWARI, M.K. *Swarm intelligence - focus on ant and particle swarm optimization*. Viena: I-Tech Education and Publishing, 2007. 532p.

CLERC, M.; KENNEDY, J. The particle swarm: Explosion, stability, and convergence in a multi-dimensional complex space. *IEEE Transactions on Evolutionary Computation*, Piscataway, v.6, n.1, p.58-73. 2002.

CLUTTER, J.C.; FORTSON, J.C.; PLENAAR, L.V.; BRISTER, G.H.; BAILEY, R.L. **Timber management: a quantitative approach**. 3ed. New York: John Wiley, 1983. 333p.

DUCHEYNE, E.I.; WULF, R.R.; BAETS, B. Single versus multiple objective genetic algorithms for solving the even-flow forest management problem. *Forest Ecology and Management*, Amsterdam, v.201, n.3-4, p.259-273. 2004.

EBERHART, R.C.; SHI, Y. Comparing inertia weights and constriction factors in particle swarm optimization. In: CONGRESS ON EVOLUTIONARY COMPUTATION, 2000, New York, USA. *Proceedings...* New York, 2000. p.84-88.

GOLDBARG, M.C.; LUNA, H.P.L. **Otimização combinatória e programação linear: modelos e algoritmos**. 2ed. Rio de Janeiro: Campus, 249 p. 2005.

GOMIDE, L.R.; ARCE, J.L.; SILVA, A.C.L. Uso do algoritmo genético no planejamento florestal considerando seus operadores de seleção. *Cerne*, Lavras, v.15, n.4, p.460-467. 2009.

GUO, Q., YU, H.; XU, A. A hybrid PSO-GD based intelligent method for machine diagnosis. *Digital Signal Processing*. v.16, n.4, p.402-418. 2006.

JOHNSON, K.N.; SCHEURMANN, H.L. Techniques for prescribing optimal timber harvest and investment under different objectives - discussion and synthesis. *Forest Science*, Washington, v.18, n.1, p.1-31, 1977.

KENNEDY, J.; EBERHART, R.C. Particle swarm optimization. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, Perth, Austrália, 1995. *Proceedings...* Perth: IEEE, 1995. p.1942-1948.

MENDES, R. **Population topologies and their influence in particle swarm performance**. 2004. 189 p. Dissertação (Mestrado em Filosofia) - Universidade do Minho, Braga, 2004.

NASCIMENTO, F.A.F. **Modelagem biométrica e planejamento florestal otimizado utilizando a meta-heurística enxame de partículas**. 2010. 99p. Dissertação (Mestrado em Ciências Florestais) - Universidade Estadual do Centro-Oeste, Riozinho, 2010.

- PEREIRA, G.W. **Aplicação da técnica de recozimento simulado em problemas de planejamento florestal multiobjetivo**. 2004. 84p. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Minas Gerais, Belo Horizonte, 2004.
- PUKKALA, T.; KURTTILA, M. Examining the performance of six heuristic optimisation techniques in different forest planning problems. *Silva Fennica*, Helsinki, v.39, n.1, p.67-80. 2009.
- R DEVELOPMENT CORE TEAM. **R: A Language and Environment for Statistical Computing**. Vienna: R Foundation for Statistical Computing, 2006. 2975p.
- RODRIGUES, F.L.; LEITE H.G.; SANTOS, H.N.; SOUZA, A.L. Soluções de problemas de planejamento florestal com restrições de inteireza utilizando Busca Tabu. *Revista Árvore*, Viçosa, v.27, n.5, p.701-713. 2003.
- RODRIGUES, F.L.; LEITE H.G.; SANTOS, H.N.; SOUZA, A.L; SILVA, G.F. Metaheurística algoritmo genético para soluções de problemas de planejamento florestal com restrições de integridade. *Revista Árvore*, Viçosa, v.28, n.2, p.233-245, 2004a.
- RODRIGUES, F.L.; LEITE H.G.; SANTOS, H.N.; SOUZA, A.L; RIBEIRO, A.A.S. R. Metaheurística simulated annealing para soluções de problemas de planejamento florestal com restrições de integridade. *Revista Árvore*, Viçosa, v.28, n.2, p.247-256, 2004b.
- SHI, Y.; EBERHART. C.A modified particle swarm optimizer. In: IEEE INTERNATIONAL CONFERENCE ON EVOLUTIONARY COMPUTATION, Piscataway, 1998. *Proceedings...* Piscataway: IEEE, 1998. p. 69-73.
- SILVA, M.L.; JACOVINE, L.A.G.; VALVERDE, S.R. **Economia florestal**. Viçosa: UFV, 178p. 2005.

Recebido em 25/12/2011

Aceito para publicação em 25/10/2012

