UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS AGRÁRIAS
DEPARTAMENTO DE COMPUTAÇÃO

GUILHERME VEZULA MATEVELI

ALGORITMOS EVOLUTIVOS PARA RESOLUÇÃO DE UM PROBLEMA DE ROTAÇÃO DE CULTURAS COM RESTRIÇÕES DE ADJACÊNCIAS

ALEGRE - ES
2014

GUILHERME VEZULA MATEVELI

ALGORITMOS EVOLUTIVOS PARA RESOLUÇÃO DE UM PROBLEMA DE ROTAÇÃO DE CULTURAS COM RESTRIÇÕES DE ADJACÊNCIAS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Geraldo Regis Mauri.

ALEGRE - ES

2014

GUILHERME VEZULA MATEVELI

ALGORITMOS EVOLUTIVOS PARA RESOLUÇÃO DE UM PROBLEMA DE ROTAÇÃO DE CULTURAS COM RESTRIÇÕES DE ADJACÊNCIAS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Aprovado em 18 de fevereiro de 2014.

BANCA EXAMINADORA

Prof. Dr. Geraldo Regis Mauri
Universidade Federal do Espírito Santo
Orientador

Prof. MSc. Antonio Almeida de Barros Junior
Universidade Federal do Espírito Santo

Prof. MSc. Juliana Pinheiro Campos Pirovani
Universidade Federal do Espírito Santo

“A menos que alguém como você se importe muito com algo, nada irá melhorar. Nada mesmo.”

Dr. Seuss

AGRADECIMENTOS

A Deus, por ter me dado todas as oportunidades de ser alguém na vida.

A meus pais, pela presença nas horas mais difíceis.

À minhas irmãs, por serem minhas conselheiras.

Aos meus sobrinhos, por estarem presentes em minha vida mesmo de longe.

Aos familiares, que sempre torceram muito por mim.

Aos amigos, por ser minha família enquanto longe de casa.

A todas as dificuldades que enfrentei, elas foram adversárias dignas e tornaram minhas vitórias muito mais saborosas.

RESUMO

O Problema de Rotação de Culturas (PRC) consiste em encontrar uma melhor programação de plantio considerando algumas restrições, ou seja, diferentes espécies são cultivadas em sucessões repetidas, na mesma área, e em uma sequência definida. O objetivo é encontrar a programação de plantio que garanta uma produção agrícola sustentável, utilizando o máximo de culturas possíveis, mas que também maximize a lucratividade. Por ser um problema complexo, o uso de métodos aproximativos é uma estratégia interessante. Portanto, a ideia principal desse trabalho consiste na aplicação de algoritmos evolutivos para resolução de uma variação do PRC, o Problema de Rotação de Culturas com Restrições de Adjacências (PRC-A). Dentre os algoritmos evolutivos, foram utilizados neste trabalho um Algoritmo Genético e um Algoritmo Memético. Para avaliar o desempenho dos métodos propostos foram realizados experimentos computacionais e os resultados obtidos foram comparados a resultados de outros trabalhos e/ou métodos existentes na literatura. Desse modo, foi possível alcançar boas soluções para o problema com o intuito de auxiliar produtores agrícolas a melhorar sua lucratividade obedecendo às restrições para uma agricultura lucrativa e sustentável.

Palavras-chave: Algoritmo Genético; Algoritmo Memético; Problema de Rotação de Culturas com Restrições de Adjacências.

EVOLUTIONARY ALGORITHMS FOR SOLVING A CROP ROTATION PROBLEM WITH ADJACENCY CONSTRAINTS

ABSTRACT

The Crop Rotation Problem (CRP) consists of finding a better planting schedule considering some restrictions, i. e., different species must be cultivated in repeated succession in the same area and in a defined sequence. The goal is finding the planting schedule to ensure a sustainable agricultural production using the maximum number of cultures for maximizing the production profit. The CRP is a complex problem and using approximation methods is an interesting strategy. Therefore, the main idea of this work is based on the application of evolutionary algorithms for solving a variation of the CRP denoted as Crop Rotation Problem with Adjacency Constraints (CRPAC). Among the evolutionary algorithms we have used a Genetic Algorithm and a Memetic Algorithm. The performance of the proposed methods was evaluated by computational experiments and the results obtained were compared against the ones reported in other works found in the literature. Thus, our methods were able to find good solutions for the problem assisting farmers to improve their production profit complying constraints for a profitable and sustainable agriculture.

Keywords: Genetic Algorithm; Memetic Algorithm, Crop Rotation Problem with Adjacency Constraints.

LISTA DE FIGURAS

Figura 1 - Fluxograma do AG. 17
Figura 2 - Pseudocódigo do Algoritmo Genético implementado. 21
Figura 3 - Pseudocódigo do Algoritmo Memético implementado. 22
Figura 4 - Pseudocódigo da Busca Local implementada. 23
Figura 5 - Exemplo do cruzamento para o PRC-A. 24
Figura 6 - Exemplo de mutação para o PRC-A. 25
Figura 7 - Relação de adjacência entre os lotes para cada instância. 25
Figura 8 - Melhor programação de plantio - AG - 10 lotes. 30
Figura 9 - Melhor programação de plantio - AG - 15 lotes. 30
Figura 10 - Melhor programação de plantio - AG - 20 lotes. 30
Figura 11 - Melhor programação de plantio - AM - 15 lotes. 31
Figura 12 - Melhor programação de plantio - AM - 20 lotes. 31
Figura 13 - Comportamento do AG e do AM (10 lotes) com restrição de tempo. 32
Figura 14 - Comportamento do AG e do AM (15 lotes) com restrição de tempo. 32
Figura 15 - Comportamento do AG e do AM (20 lotes) com restrição de tempo. 33
Figura 16 - Comportamentos do AG e do AM (10 lotes) em 10000 gerações. 33
Figura 17 - Comportamentos do AG e do AM (15 lotes) em 10000 gerações. 34
Figura 18 - Comportamentos do AG e do AM (20 lotes) em 10000 gerações. 34

LISTA DE TABELAS

Tabela 1 - Área (em ha) dos lotes. 26
Tabela 2 - Culturas disponíveis para o plantio. 26
Tabela 3 - Valores dos parâmetros do AG e AM após a calibração. 28
Tabela 4 - Pesos adotados para as violações nas restrições. 29
Tabela 5 - Resultados obtidos por meio de 100 execuções do AG proposto. 29
Tabela 6 - Resultados obtidos por meio de 100 execuções do AM proposto. 29
Tabela 7 - Comparação das soluções obtidas pelo AG com a literatura. 35
Tabela 8 - Comparação das soluções obtidas pelo AM com a literatura. 35
Tabela 9 - Comparação dos tempos computacionais médios obtidos. 35

SUMÁRIO

1 INTRODUÇÃO 11
1.1 O PROBLEMA E SUA IMPORTÂNCIA 11
1.2 OBJETIVOS 13
1.2.1 Objetivo geral 13
1.2.2 Objetivos específicos 13
2 REVISÃO DA LITERATURA 14
2.1 MODELO MATEMÁTICO DO PRC-A 14
2.2 ALGORITMO GENÉTICO 16
2.2.1 Geração da população inicial 17
2.2.2 Avaliação da população 18
2.2.3 Seleção dos pais candidatos 18
2.2.4 Cruzamento (*Crossover*) 18
2.2.5 Mutação 18
2.2.6 Definição da população sobrevivente 18
2.3 ALGORITMO MEMÉTICO 18
2.4 TRABALHOS CORRELATOS 19
3 METODOLOGIA 21
3.1 ALGORITMOS IMPLEMENTADOS 21
3.2 EXPERIMENTOS COMPUTACIONAIS 25
4 RESULTADOS OBTIDOS 28
4.1 RESULTADOS COMPUTACIONAIS 28
4.2 COMPARAÇÃO COM A LITERATURA 34
5 CONCLUSÕES 37
REFERÊNCIAS BIBLIOGRÁFICAS 38

1 INTRODUÇÃO

Com a urbanização das cidades e a diminuição dos recursos naturais, as fazendas de produção agrícola buscam maneiras de melhorar o manejo de culturas visando à lucratividade.

A agricultura convencional possui dois objetivos que relacionam entre si: a maximização da produção e a maximização do lucro. Práticas são buscadas sem cuidar das consequências e sem considerar a dinâmica ecológica dos agroecosistemas. Entre elas, duas práticas básicas podem ser citadas: cultivo intensivo do solo e a monocultura (GLIESSMAN, 2005).

Infelizmente, o cultivo intensivo da agricultura convencional pode prejudicar o solo e assim diminuir a qualidade do produto. Desse modo, uma maneira de buscar a lucratividade é a produção agrícola a partir dos conceitos de agroecologia e agricultura sustentável.

1.1 O PROBLEMA E SUA IMPORTÂNCIA

Para evitar o cultivo intensivo que degrada o solo e a monocultura que aumenta a utilização de defensivos agrícolas, é imprescindível que haja uma expansão da policultura a partir dos conceitos de agroecologia, que leva em consideração a preservação ambiental das áreas de plantio, o controle biológico de pragas e os ganhos econômicos (ALIANO FILHO, 2012).

Tendenciosamente, com a utilização da policultura para a produção agrícola, é necessária a utilização da rotação de culturas. A rotação de culturas é uma das principais estratégias para a diversificação de plantação de vegetais em uma mesma área durante um determinado tempo. Especificamente, a rotação de culturas é um sistema no qual diferentes espécies são cultivadas em sucessões repetidas, na mesma área, e em uma sequência definida (ALTIERI, 2012).

Segundo Arf et al. (1999), a rotação de culturas melhora as condições físicas, químicas e biológicas do solo, além de estabelecer o controle de plantas daninhas, de doenças e pragas, a reposição de restos orgânicos e a proteção do solo contra a ação dos agentes climáticos.

Dentre os critérios para uma agricultura sustentável, Arf et al. (1999), Altieri (2012) e Gliessman (2005) propõem as principais restrições para a utilização da rotação de culturas: (a) separar os cultivos da mesma família botânica em lotes adjacentes; (b) não cultivar uma mesma família botânica de plantas em uma mesma área consecutivamente; (c) garantir o plantio de culturas leguminosas (adubação verde); (d) determinar o descanso do solo (período de pousio); (e) respeitar o período de plantio de cada cultura; e (f) atender os ciclos que cada cultura necessita para o cultivo.

Combinando cultivos de famílias botânicas diferentes, a ação de pestes é reduzida e a estabilidade da produção aumenta perante os impactos ambientais. Já a alternância entre diferentes espécies botânicas em uma mesma área reduz o empobrecimento do solo, garantindo um melhor aproveitamento dos recursos naturais da terra. As leguminosas ajudam a descompactar o solo e a simbiose com bactérias fixadoras de hidrogênio (ARF et al., 1999). Esse processo evita erosão do solo, melhora sua estrutura e fertilidade e diminui as pragas e a vegetação espontânea. No período de pousio, permite-se que a vegetação cresça livremente para recuperação de seus nutrientes naturais. Por fim, para garantir que as culturas sejam plantadas e colhidas na época correta, o ciclo e a época de plantio devem ser respeitados.

Considerando então as características apresentadas, surge um problema combinatorial conhecido como Problema de Rotação de Culturas (PRC) que, segundo Santos et al. (2007), consiste em um sistema de plantio sequencial de espécies vegetais sobre a mesma área, de modo que esta sequência se repita em um determinado período de tempo.

Em meio às diferentes abordagens do PRC, tem-se o Problema de Rotação de Culturas com Restrições de Adjacências (PRC-A) e o Problema de Rotação de Culturas com Atendimento de Demanda (PRC-D). O PRC-A consiste na inclusão de restrições de adjacências, sendo que culturas da mesma família botânica não podem ser plantadas em lotes adjacentes. O PRC-D, por sua vez, faz com a produção de todas as culturas atendam uma determinada demanda, em um intervalo que ela é solicitada (ALIANO FILHO, 2012).

Mais detalhes sobre essas abordagens são apresentados em Santos (2009), Santos et al. (2010), Costa et al. (2011) e Santos et al. (2011). Neste trabalho, o modelo escolhido para estudo é o PRC-A, e o objetivo principal consiste em maximizar o lucro em um determinado período de tempo, respeitando as restrições citadas anteriormente.

1.2 OBJETIVOS

A seguir, são apresentados os objetivos gerais e específicos deste trabalho.

1.2.1 Objetivo geral

O objetivo geral desse trabalho consiste na aplicação das meta-heurísticas Algoritmo Genético (AG) e Algoritmo Memético (AM) para resolução do Problema de Rotação de Culturas com Restrições de Adjacências (PRC-A). Assim, buscou-se por novas alternativas para a resolução do problema.

1.2.2 Objetivos específicos

a) Implementar as adaptações necessárias para o AG e o AM.

b) Aplicar o AG e o AM para a resolução de instâncias para o PRC-A encontradas na literatura.

c) Avaliar o desempenho dos métodos por meio dos resultados computacionais obtidos quando comparados a resultados de outros trabalhos e/ou métodos existentes na literatura.

d) Apresentar uma alternativa para auxiliar produtores agrícolas em suas tomadas de decisões.

Os demais capítulos deste trabalho estão assim divididos: o Capítulo 2 apresenta uma revisão bibliográfica. O Capítulo 3 descreve a metodologia utilizada para o desenvolvimento deste trabalho e no Capítulo 4 são apresentados os resultados obtidos. Por fim, o Capítulo 5 apresenta as conclusões obtidas.

2 REVISÃO DA LITERATURA

2.1 MODELO MATEMÁTICO DO PRC-A

Nessa seção, é descrita a formulação matemática proposta por Aliano Filho (2012), que é a base deste trabalho para representação do Problema de Rotação de Culturas com Restrições de Adjacências (PRC-A).

Considerando uma rotação de culturas de T anos divididos em M períodos, define-se uma *programação de rotação de culturas* de tamanho M como um calendário de plantio e cultivo na unidade de tempo (dia, semana, mês, bimestre, etc.).

A programação de rotação consiste em decidir quais culturas deverão ser plantadas, em que período da rotação e os lotes onde estas deverão ser cultivadas. Pode associar-se um grafo $G(V,A)$ à área de cultivo dividida em lotes, sendo que V é o conjunto dos lotes e A é o conjunto de arestas de modo que o par $(u,v) \in$ A se e somente se os lotes u e v são adjacentes.

Entre os possíveis critérios, as mesmas restrições técnicas e de base sustentável foram seguidas. Alguns parâmetros e variáveis foram considerados para descrever o conjunto de programações de rotações de culturas que respeitam as restrições de (a) até (f), descritas na Seção 1.1, a saber:

- M – número de períodos de mesma duração (unidade de tempo pré-estabelecida) da rotação;
- N_f – número de famílias de plantas botânicas;
- N – número de culturas pertencentes à N_f famílias botânicas;
- L – número de lotes para plantio;
- C – conjunto de culturas que podem ser selecionadas para fins comerciais, excluindo as culturas de adubação verde;
- A – conjunto de culturas que podem ser alocadas para adubação verde (leguminosas);
- F_p – conjunto de culturas da família botânica p, tal que $p = 1,...,N_f$;
- t_i – ciclo de vida da cultura i, incluindo os períodos estimados de preparação do solo e colheita;

- $I_i = [C_i, T_i]$ – intervalo de semeadura da cultura *i*, tal que C_i é o período mais cedo e T_i é o período mais tarde;
- l_i – lucratividade (R$/ha) da cultura *i* nos t_i períodos;
- *T* – duração da programação, igual para todos os lotes;
- $area_k$ – área do lote *k* em hectares (ha);
- S_k – o conjunto dos lotes adjacentes ao lote *k*.
- Por simplicidade, o pousio é considerado como sendo a cultura indicada por $n = N + 1$. A variável decisória é representada da seguinte forma:
- $x_{ijk} = 1$, se a cultura *i* é plantada no período *j* no lote *k*, e 0 caso contrário, para $i = 1,...,N+1$, $j \in I_i$ e $k = 1,...,L$.

A formulação matemática para maximizar a lucratividade da produção com o intuito de atender a todas as restrições impostas é descrita a seguir:

$$\textbf{Maximizar } z = \sum_{i \in C} \sum_{j \in I_i} \sum_{k=1}^{L} area_k l_i x_{ijk} \tag{1}$$

Sujeito a

$$\sum_{i \in F_p} \sum_{r=0}^{t_i - 1} \sum_{v \in S_k} x_{i(j-r)v} \leq L \left(1 - \sum_{i \in F_p} x_{ijk} \right), p = 1, \dots, N_f, j = 1, \dots, M, k = 1, \dots, L \tag{2}$$

$$\sum_{i \in F_p} \sum_{r=0}^{t_i} x_{i(j-r)k} \leq 1, p = 1..N_f, j = 1, \dots, M, k = 1, \dots, L \tag{3}$$

$$\sum_{i=1}^{N+1} \sum_{r=0}^{t_i - 1} x_{i(j-r)k} \leq 1, j = 1, \dots, M, k = 1, \dots, L \tag{4}$$

$$\sum_{j=1}^{M} x_{njk} \geq 1, k = 1, \dots, L \tag{5}$$

$$\sum_{i \in A} \sum_{j=1}^{M} x_{ijk} \geq 1, k = 1, \dots, L \tag{6}$$

$$x_{ijk} \in \{0,1\}, i = 1, \dots, N + 1, j \in I_i, k = 1, \dots, L \tag{7}$$

A função objetivo (1) provê a maximização da lucratividade da rotação a ser realizados nos *L* lotes, nos *M* períodos, com as culturas comerciais pertencentes ao conjunto *C*.

A restrição (2) proíbe que culturas da mesma família botânica sejam cultivadas em lotes adjacentes ao mesmo tempo. A restrição (3) impede que culturas de mesma família botânica possam ser cultivadas consecutivamente dentro de um mesmo lote.

A restrição (4) é referente ao ciclo de vida de uma cultura, ou seja, impede que uma cultura seja plantada sem seu ciclo de vida completo.

A existência de pelo menos um pousio em cada lote durante programação de rotação é garantida pela restrição (5). Para o pousio, não é considerada nenhuma restrição de adjacência, ou seja, o pousio não é tratado como uma cultura. A restrição (6) se refere ao plantio de pelo menos uma cultura de adubação verde (leguminosa), e a restrição (7) garante que as variáveis de decisão sejam binárias.

2.2 ALGORITMO GENÉTICO

O AG foi proposto por Holland (1975) e posteriormente Goldberg (1989) foi um dos autores que contribuíram para a consolidação de seu trabalho. O AG é inspirado no princípio Darwiniano da evolução das espécies e na genética. É originário da analogia entre a reprodução de uma estrutura por meio de um vetor de componentes, construindo um cromossomo artificial.

O AG é um algoritmo evolutivo por ser considerado um método de otimização não-determinístico baseado nos princípios da evolução biológica natural e por possuir uma versatilidade e adaptabilidade em problemas complexos e combinatórios (CHIPPERELD; FLEMING, 1996).

Goldberg (1989) utilizou um vocabulário “emprestado” da genética natural para descrever os termos utilizados no AG:

- Cromossomo: representa as informações relativas às variáveis do problema.
- Indivíduo: solução representada pelo cromossomo.
- Gene: unidade básica do cromossomo (posição do cromossomo).
- Alelo: possível valor que cada gene pode assumir.
- População: conjunto de cromossomos ou indivíduos.
- Geração: iteração em que o AG está sendo executado.
- Operações gênicas: operações realizadas nos cromossomos.

- Aptidão ou *fitness*: informação numérica de cada cromossomo na população que indica sua “qualidade”.

A Figura 1 apresenta um fluxograma com os principais componentes do AG e a forma com que se relacionam. Inicialmente, são gerados indivíduos (soluções) que irão formar a população inicial. O *fitness* é calculado para cada indivíduo e a população é avaliada. Para cada iteração do AG, dois indivíduos pais da população são selecionados para um *crossover*. O objetivo do *crossover* é gerar novos indivíduos filhos pela troca de *genes* entre os pais (GOLDBARG; LUNA, 2005). Os novos indivíduos são inseridos na população e o *fitness* dos novos indivíduos é calculado. Assim, na iteração/geração atual é definida a população sobrevivente. Caso o critério de parada seja satisfeito, os melhores indivíduos são listados e, caso contrário, é realizada uma nova iteração.

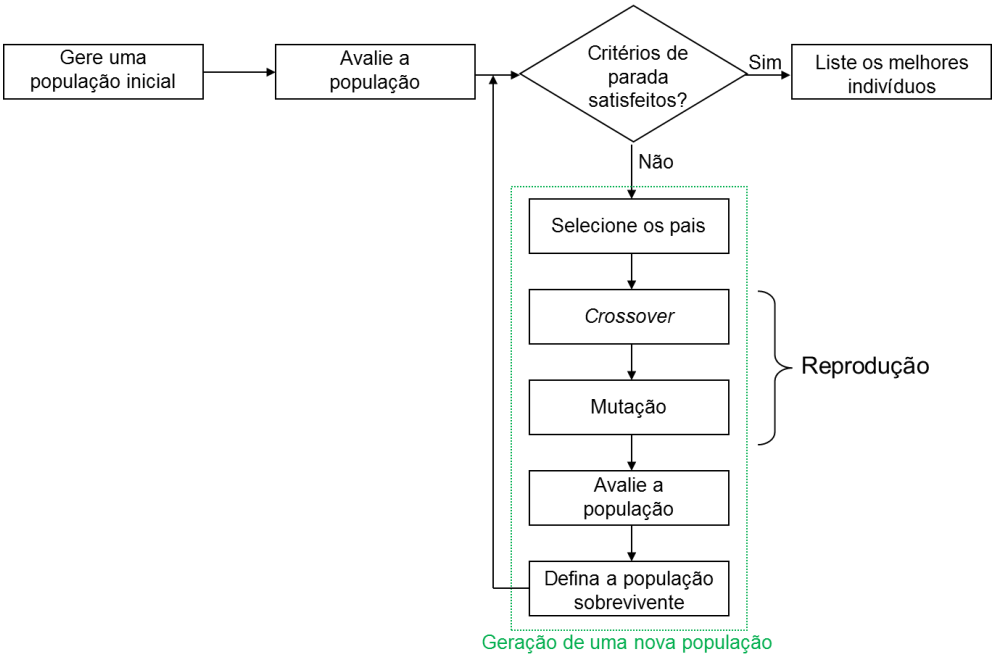

Figura 1 - Fluxograma do AG.

Fonte: Souza (2012).

2.2.1 Geração da população inicial

O gerador de população deve ser um método capaz de gerar indivíduos distintos de forma aleatória ou por meio de outro método específico. O desenvolvimento do AG está associado à herança genética e, portanto, a população inicial influencia em seu desempenho (GOLDBARG; LUNA, 2005).

2.2.2 Avaliação da população

Segundo Goldbarg e Luna (2005), é uma função de avaliação que classifica os indivíduos de uma população. Os indivíduos da população podem ser classificados por sua performance e/ou pela avaliação do *fitness*. O *fitness* de uma solução é um indicador natural para a avaliação de qualidade de um cromossomo.

2.2.3 Seleção dos pais candidatos

Indivíduos com melhor *fitness* terão maior probabilidade de reprodução do que os piores, ou seja, mais sucesso nas competições de sobrevivência. É importante perceber a aleatoriedade da seleção dos indivíduos pais para manter a diversidade da população (GOLDBARG; LUNA, 2005).

2.2.4 Cruzamento (*Crossover*)

São operadores associados à reprodução que utilizam um ou mais indivíduos para reprodução. Esses operadores utilizam dois indivíduos para cruzamento, sendo que os principais existentes na literatura são: crossover uni-ponto, multiponto, uniforme, *operator cycle*, segmentado entre outros (GOLDBARG; LUNA, 2005).

2.2.5 Mutação

Segundo Goldberg (1989), a mutação é responsável por reproduzir um novo material genético em indivíduos existentes. Sua principal importância é impedir que população se estabilize à baixa diversidade e à pressão seletiva.

2.2.6 Definição da população sobrevivente

Os melhores indivíduos se propagam através das gerações para que sejam aperfeiçoados por meio de cruzamentos sucessivos gerando cada vez filhos melhores (GOLDBARG; LUNA, 2005). Nessa parte do algoritmo, os novos indivíduos entram na população e essa população é ordenada a partir do *fitness*, permanecendo apenas os melhores indivíduos.

2.3 ALGORITMO MEMÉTICO

Além do Algoritmo Genético (AG), neste trabalho também foi implementado um outro algoritmo evolutivo, o Algoritmo Memético (AM).

O Algoritmo Memético (AM) é um algoritmo evolutivo que pode ser visto como um AG que utiliza uma Busca Local (BL) como procedimento para aperfeiçoar a busca por indivíduos melhores (MOSCATO, 1999).

O AM utiliza operadores evolutivos para encontrar regiões promissoras e aplica um procedimento de BL para intensificar a investigação nessas regiões (MAURI et al., 2011). A BL geralmente é aplicada sobre os indivíduos gerados pelo cruzamento, após a realização da mutação.

2.4 TRABALHOS CORRELATOS

Na literatura, há diversos trabalhos sobre o Problema de Rotação de Culturas (PRC) e suas variações. A seguir, é apresentada em ordem cronológica uma revisão bibliográfica sobre alguns dos principais trabalhos sobre este problema.

Santos et al. (2007) apresentaram um dos principais trabalhos referente ao PRC, no qual foi proposto um modelo de otimização binário para o problema. Foram utilizados dados reais de 28 culturas, com diferentes famílias, ciclo e época de plantio. O modelo proposto foi resolvido pelo *solver* CPLEX 7.5 com instâncias de 1 a 10 lotes.

Santos (2009) propôs métodos exatos e heurísticos para o PRC, como exemplo: heurísticas construtivas, geração de colunas, *branch-and-bound*, *branch-and-price,* entre outros. Alguns dos algoritmos propostos foram capazes de provar rapidamente a otimalidade para os problemas testes considerados. Todos os algoritmos foram implementados na linguagem C++ e os modelos lineares inteiros foram resolvidos no CPLEX 11.1 considerando 28 culturas com 4, 8, 12, 16 e 20 lotes.

Em seguida, Santos et al. (2011) tratou o PRC-A como um problema de otimização binário utilizando uma heurística baseada em geração de colunas. Nesse caso, foi usada a decomposição de *Dantzig-Wolfe* com a heurística em questão para atingir soluções ótimas mais rapidamente. Testes computacionais foram realizados utilizando 28 culturas distintas com 1, 3, 5, 7 e 9 lotes.

Para finalizar, Aliano Filho (2012) apresentou soluções para o PRC-A e PRC-D utilizando diferentes meta-heurísticas como Algoritmo Genético (AG), *Simulated Annealing* (SA), SA+AG e Algoritmo Memético (AM). Para o problema em questão, o

objetivo considerado consiste na maximização da receita (lucratividade) e, portanto, foi considerado o lucro/lote de cada cultura e a área do lote como variáveis do problema. Para cada uma das quatro meta-heurísticas propostas, foram utilizados dados intrínsecos às culturas, sendo estes utilizados para formação de três instâncias diferentes, com 10, 15 e 20 lotes, considerando uma estrutura geográfica hipotética. Essas instâncias são formadas por 30 culturas distintas, sendo 25 comerciais, 4 destinadas à adubação verde e um pousio.

3 METODOLOGIA

A metodologia utilizada neste trabalho consiste na aplicação das meta-heurísticas Algoritmo Genético (AG) e Algoritmo Memético (AM) para resolução do Problema de Rotação de Culturas com Restrições de Adjacências (PRC-A).

3.1 ALGORITMOS IMPLEMENTADOS

Nas Figuras 2, 3 e 4 são apresentados os pseudocódigos do AG, do AM e da BL desenvolvidos neste trabalho. Os algoritmos (AG e AM) possuem cinco parâmetros: *Tem* indica o tempo limite de parada dos algoritmos, ou seja, ao atingir um tempo determinado os algoritmos são finalizados; *Pop* define o número de indivíduos da população; *Mel* é a probabilidade em que um indivíduo com melhor *fitness* é selecionado para o cruzamento; *Cro* indica o número de cruzamentos entre os indivíduos selecionados; e *Mut* é a probabilidade em que um filho sofre uma mutação.

```
AG(Tem, Pop, Mel, Cro e Mut) FAÇA
P ← Uma população inicial com Pop indivíduos;
nP ← ∅;      g ← 0;      tE ← 0;
ENQUANTO(tE < Tem) FAÇA
    g ← g + 1;
    PARA c DE 1 ATÉ Cro FAÇA
        COM probabilidade = Mel ⇒ p1 ← um indivíduo aleatório de P;
        p2 ← um indivíduo aleatório de P;
        CROSSOVER(p1, p2);
        SE (probabilidade ≤ Mut) ENTÃO f1 ← MUTAÇÃO(f1); FIM-SE;
        SE (probabilidade ≤ Mut) ENTÃO f2 ← MUTAÇÃO(f2); FIM-SE;
        nP ← nP ∪ {f1, f2};          P ← P* ⇒ P* ⊆ {P ∪ nP};
        P ← ORDENAÇÃO(P);            nP ← ∅;
    FIM-PARA;
    ENQUANTO(Pop < |P|) FAÇA
        p ← pior indivíduo de P;
        P ← P* ⇒ P* ⊆ {P − p};
    FIM-ENQUANTO;
FIM-ENQUANTO;
I* ← Melhor indivíduo de P;
RETORNAR(I*);
```

Figura 2 - Pseudocódigo do Algoritmo Genético implementado.

A diferença entre o AG e AM implementados é a aplicação da Busca Local nos novos indivíduos da população com intuito de aperfeiçoar a busca por indivíduos melhores. Na Figura 3, a Busca Local (BL) é apresentada nas linhas 12 e 13, após a mutação dos novos indivíduos.

```
1.  AM (Tem, Pop, Mel, Cro e Mut) FAÇA
2.  P ← Uma população inicial com Pop indivíduos;
3.  nP ← ∅;   g ← 0;   tE ← 0;
4.  ENQUANTO (tE < Tem) FAÇA
5.      g ← g + 1;
6.      PARA c DE 1 ATÉ Cro FAÇA
7.          COM probabilidade = Mel ⇒ p1 ← um indivíduo aleatório de P;
8.          p2 ← um indivíduo aleatório de P;
9.          CROSSOVER(p1, p2);
22.         SE (probabilidade ≤ Mut) ENTÃO f1 ← MUTAÇÃO(f1); FIM-SE;
23.         SE (probabilidade ≤ Mut) ENTÃO f2 ← MUTAÇÃO(f2); FIM-SE;
10.         BUSCALOCAL(f1);
11.         BUSCALOCAL(f2);
12.         nP ← nP ∪ {f1, f2};          P ← P* ⇒ P* ⊆ {P ∪ nP};
13.         P ← ORDENAÇÃO(P);            nP ← ∅;
14.     FIM-PARA;
15.     ENQUANTO (Pop < |P|) FAÇA
16.         p ← pior indivíduo de P;
17.         P ← P* ⇒ P* ⊆ {P − p};
18.     FIM-ENQUANTO;
19. FIM-ENQUANTO;
20. I* ← Melhor indivíduo de P;
21. RETORNAR (I*);
```

Figura 3 - Pseudocódigo do Algoritmo Memético implementado.

O algoritmo da BL implementada é apresentado na Figura 4. A BL possui o parâmetro *Fil* que indica o novo indivíduo gerado pelo cruzamento. Seu funcionamento é simples, o novo indivíduo da população é clonado e, a partir disso, é realizado um movimento de troca em alguns lotes desse novo indivíduo, ou seja, um lote é selecionado de forma aleatória e a partir dele até o último lote é realizado o movimento de troca. Nesse movimento de troca todas as culturas são testadas, uma de cada vez. Desta forma, para cada cultura, um mês aleatório é selecionado e a cultura é inserida a partir desse mês. Se o *fitness* do novo indivíduo clonado for melhor que do original, ele é substituído.

```
BUSCALOCAL(Fil) FAÇA
F* ← CLONE(Fil);
L ← Um lote aleatório;
Lot ← Número de lotes do problema
Cul ← Número total de culturas
PARA l DE L ATÉ Lot FAÇA
    PARA c DE 1 ATÉ Cul FAÇA
        SE (MOVIMENTOTROCA(F*, l, c)) ENTÃO
            SE (MELHOROU(F*, Fil)) ENTÃO
                Fil ← CLONE(F*);
            FIM-SE;
        FIM-SE;
    FIM-PARA;
FIM-PARA;
```

Figura 4 - Pseudocódigo da Busca Local implementada.

A função objetivo utilizada no AG e AM é descrita pela Equação (8), que trata o problema de forma relaxada, utilizando pesos para penalizar a violação das restrições do modelo. Para cada restrição violada na programação de plantio, um peso em unidades é subtraído da lucratividade total obtida pela Função Objetivo (FO).

$$\sum_{i \in C} \sum_{j \in I_i} \sum_{k=1}^{L} area_k l_i x_{ijk} - \begin{bmatrix} (nCE \times pCE) + (nLA \times pLA) + (nPC \times pPC) + \\ (nAV \times pAV) + (nLP \times pLP) + (nCI \times pCI) \end{bmatrix} \quad (8)$$

Na FO, *nCE* indica o número e *pCE* o peso de culturas plantadas fora da época de plantio; *nLA* representa o número e *pLA* o peso de lotes adjacentes que possuem culturas de mesma família botânica; *nPC* indica o número de vezes em que culturas de mesma família botânica são plantadas consecutivamente em um mesmo lote e *pPC* é a penalidade dada à essa incidência; *nAV* indica o número e *pAV* o peso de lotes que não possuem adubação verde durante o calendário de rotação; *nLP* denota o número de lotes que não possuem pousios durante o calendário de rotação e *pLP* o peso dado; e por fim, *nCI* indica o número e *pCI* a penalidade de culturas plantadas com o ciclo de vida incorreto.

Neste trabalho, a população inicial é gerada de forma aleatória. Um indivíduo é representado por uma matriz na qual as linhas representam os Lotes e as colunas os Meses de plantio da cultura. Considera-se o Gene de um indivíduo a posição da

matriz representada por [*l*][*m*], sendo *l* o Lote e *m* o Mês de plantio. O valor atribuído ao Gene (Alelo) é a identificação da cultura (coluna Id. da Tabela 2). Na Figura 2 é possível visualizar a representação de uma solução.

Para a avaliação da população o *fitness* é calculado de acordo com a função objetivo (Equação (8)). Para cada indivíduo, sua FO é calculada e sua aptidão é definida por essa equação. Já na seleção dos pais candidatos, é utilizada a seleção do tipo Elite, na qual um dos pais será selecionado entre os melhores indivíduos e outro selecionado aleatoriamente.

Neste trabalho, o cruzamento utilizado foi o *crossover* multiponto. Esse cruzamento foi escolhido por ter demostrado soluções melhores do que o uniponto. Para isso, a partir dos pais *p1* e *p2* selecionados para geração dos filhos *f1* e *f2*, dois lotes são selecionados aleatoriamente. Do lote inicial até o primeiro lote selecionado os Genes de *p1* vão para *f1* e os Genes de *p2* para *f2*. Depois do primeiro lote selecionado até o segundo lote selecionado, os Genes de *p1* vão para *f2* e os Genes de *p2* para *f1* e, por fim, após o segundo lote selecionado até o último lote repete-se o primeiro passo, *p1* para *f1* e *p2* para *f2*. A Figura 5 apresenta um exemplo de cruzamento, considerando que o cromossomo possui 5 lotes e 12 meses para plantio.

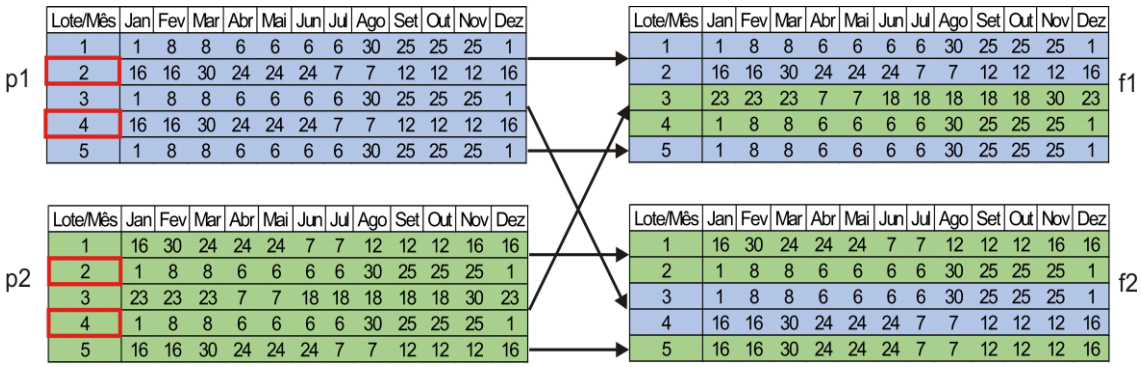

Figura 5 - Exemplo do cruzamento para o PRC-A.

A mutação é realizada após o cruzamento. A partir dos filhos gerados, com uma determinada probabilidade, a mutação é aplicada. Neste trabalho, um lote é selecionado aleatoriamente e todos os Genes desse indivíduo têm seus valores modificados, também de forma aleatória (ver Figura 6).

Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	16	16	30	24	24	24	7	7	12	12	12	16
2	1	8	8	6	6	6	6	30	25	25	25	1
3	23	23	23	7	7	18	18	18	18	18	30	23
4	16	16	30	24	24	24	7	7	12	12	12	16
5	1	8	8	6	6	6	6	30	25	25	25	1

Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	16	16	30	24	24	24	7	7	12	12	12	16
2	1	8	8	6	6	6	6	30	25	25	25	1
3	23	23	23	7	7	18	18	18	18	18	30	23
4	23	23	23	7	7	18	18	18	18	18	30	23
5	1	8	8	6	6	6	6	30	25	25	25	1

Figura 6 - Exemplo de mutação para o PRC-A.

3.2 EXPERIMENTOS COMPUTACIONAIS

Para realização de experimentos computacionais com o AG e o AM propostos, foram utilizadas três instâncias (problemas testes ou *benchmarks*), com 10, 15 e 20 lotes, respectivamente.

Essas instâncias são as mesmas apresentadas por Aliano Filho (2012), o qual aplicou diferentes métodos para resolução do PRC-A. Essas instâncias são formadas por 29 culturas diferentes, sendo 25 comerciais e 4 destinadas à adubação verde, além do pousio (cultura 30).

A Figura 7 apresenta o esquema de adjacência entre os lotes para as três instâncias consideradas.

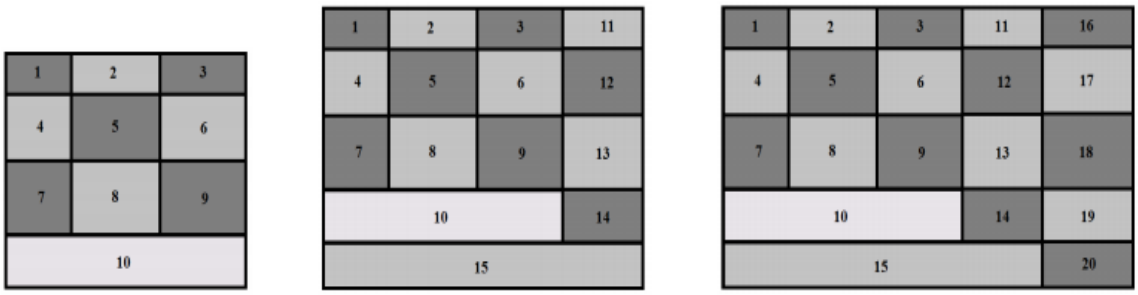

Figura 7 - Relação de adjacência entre os lotes para cada instância.
Fonte: Aliano Filho (2012).

A área de cada lote (em ha) é descrita na Tabela 1, e a Tabela 2 apresenta os dados referentes às culturas disponíveis para o plantio, sendo: nome da cultura, família botânica, época de plantio, ciclo de vida (em meses) e suas respectivas lucratividades (em R$).

Tabela 1 - Área (em ha) dos lotes.

Fonte: Aliano Filho (2012).

Lote	Área	Lote	Área
1	1,50	11	2,00
2	2,00	12	3,00
3	2,00	13	4,00
4	2,25	14	3,00
5	3,00	15	5,50
6	3,00	16	2,50
7	3,00	17	3,75
8	4,00	18	5,00
9	4,00	19	3,75
10	8,25	20	2,50

Tabela 2 - Culturas disponíveis para o plantio.

Fonte: Aliano Filho (2012).

Id	Cultura	Família	Época de Plantio		Ciclo de vida	Lucratividade (R$)
			Inicio	Fim		
1	Alface	*Compositae*	Ano todo		2	300
2	Almeirão	*Compositae*	Ano todo		4	150
3	Couve	*Brassicaceae*	Março	Junho	3	300
4	Brócolis	*Brassicaceae*	Fevereiro	Junho	7	400
5	Repolho	*Brassicaceae*	Fevereiro	Junho	4	400
6	Couve-Flor	*Brassicaceae*	Março	Junho	4	900
7	Beterraba	*Chenopodiaceae*	Março	Julho	2	810
8	Espinafre	*Chenopodiaceae*	Fevereiro	Julho	2	600
9	Abobrinha	*Cucurbitaceae*	Agosto	Março	4	400
10	Moranga	*Cucurbitaceae*	Setembro	Janeiro	5	200
11	Pepino	*Cucurbitaceae*	Ano todo		4	450
12	Melancia	*Cucurbitaceae*	Agosto	Outubro	3	900
13	Alho	*Liliaceae*	Março	Abril	6	810
14	Cebola	*Liliaceae*	Março	Junho	4	430
15	Quiabo	*Malvaceae*	Agosto	Março	7	710
16	Milho	*Gramineae*	Agosto	Abril	3	350
17	Aveia	*Gramineae*	Março	Maio	6	350
18	Tomate	*Solanaceae*	Ano todo		5	810
19	Pimentão	*Solanaceae*	Ano todo		5	550
20	Batata	*Solanaceae*	Agosto	Outubro	3	240
21	Cenoura	*Umbelliferae*	Março	Julho	4	620
22	Salsinha	*Umbelliferae*	Setembro	Março	6	400
23	Feijão-Vagem	*Leguminosae*	Agosto	Abril	4	750
24	Ervilha	*Leguminosae*	Março	Abril	3	830
25	Feijão	*Leguminosae*	Agosto	Setembro	3	720
26	Crotalália	*Leguminosae*	Setembro	Dezembro	3	-
27	Feijão-de-Porco	*Leguminosae*	Setembro	Dezembro	3	-
28	Mucuna	*Leguminosae*	Setembro	Janeiro	3	-
29	Ervilha Peluda	*Leguminosae*	Março	Junho	4	-
30	Pousio	-	Ano todo		1	-

Para cada instância do PRC-A, o AG e o AM foram executados 100 vezes, com o intuito de se obter as seguintes informações:

- *Melhor Função Objetivo (MFO)*: o valor do melhor indivíduo (melhor solução) gerado nas 100 execuções do AG ou do AM.

- *FO média (MED)*: a média aritmética das melhores FOs encontradas nas 100 execuções do AG ou do AM.

- *Desvio padrão (DP)*: calculado da seguinte forma:

$$DP = (((MFO - MED) \,/\, MFO) * 100) \tag{9}$$

- *Tempo médio (TM)*: a média aritmética dos tempos computacionais para encontrar a MFO (em segundos) para as 100 execuções do AG ou do AM.

- *Geração da melhor Função Objetivo (GE)*: em qual geração que a MFO foi encontrada.

4 RESULTADOS OBTIDOS

A seguir, são apresentados os resultados obtidos por meio dos experimentos computacionais. Além disso, também é apresentada uma comparação dos resultados obtidos em relação aos obtidos por outros métodos encontrados na literatura.

4.1 RESULTADOS COMPUTACIONAIS

Para calibração dos parâmetros do AG e AM, as três instâncias (10, 15 e 20 lotes) foram resolvidas diversas vezes, variando sempre os valores de cada parâmetro. Assim, um parâmetro recebeu vários valores enquanto os demais continuaram com valores fixos. Esse processo foi repetido até que se adequassem a todas as instâncias, ou seja, obtivesse soluções viáveis. Após definir o valor mais adequado para um parâmetro, outro parâmetro foi escolhido pelo mesmo processo, até que todos os fossem definidos. Ao final da calibração, os parâmetros foram definidos segundo apresentado na Tabela 3.

Tabela 3 - Valores dos parâmetros do AG e AM após a calibração.

Parâmetro	**Descrição**	**Valor**
Tem	Tempo limite de parada dos algoritmos	300 segundos
Pop	Tamanho limite da população	69 indivíduos
Mel	Probabilidade de escolher o melhor individuo para cruzamento	6%
Cro	Número de cruzamentos	96 vezes
Mut	Probabilidade de mutação	27%

Seguindo um processo análogo ao adotado para a calibração dos parâmetros do AG e do AM, os valores para os pesos utilizados na função objetivo (Equação (8)) foram definidos. A Tabela 4 apresenta os melhores valores encontrados. Esses pesos foram atribuídos empiricamente, iniciando todos com o mesmo valor, igual a 10000. O AG foi executado para as 3 instâncias (10, 15 e 30 lotes) e, enquanto as soluções obtidas eram inviáveis, o peso era aumentado nas restrições violadas. Isso foi realizado até que apenas soluções viáveis fossem obtidas. Feito isso, os mesmos valores para os pesos foram utilizados tanto no AG quanto no AM.

Tabela 4 - Pesos adotados para as violações nas restrições.

Pesos	Valor (unidades)
pCE	10000
pLA	30000
pPC	10000
pAV	89000
pLP	10000
pCI	10000

Após a calibração dos parâmetros e dos pesos, o AG e o AM foram executados 100 vezes para cada instância. Os resultados obtidos pelo AG e o AM são apresentados nas Tabelas 5 e 6, respectivamente. Nessas tabelas, são apresentados, para cada instância, o valor da melhor solução (MFO) a solução média (MED), o desvio (DP), o tempo médio para encontrar a melhor solução (TM) e a geração em que a melhor solução foi encontrada (GE).

Tabela 5 - Resultados obtidos por meio de 100 execuções do AG proposto.

Instância	MFO (R$)	MED (R$)	DP (%)	TM (seg.)	GE
10 Lotes	88295,00	83347,50	5,60	23,39	9243
15 Lotes	134440,00	127550,00	5,12	63,35	20391
20 Lotes	180575,00	171822,50	4,95	110,59	27995

Tabela 6 - Resultados obtidos por meio de 100 execuções do AM proposto.

Instância	MFO (R$)	MED (R$)	DP (%)	TM (seg.)	GE
10 Lotes	88295,00	83397,50	5,55	177,03	2415
15 Lotes	133092,50	126793,75	4,67	250,95	4506
20 Lotes	178975,00	168286,25	5,97	276,23	1904

Observa-se nas Tabelas 5 e 6 que, em média, as soluções do Algoritmo Genético (AG) foram melhores que as obtidas pelo Algoritmo Memético (AM). O motivo desse resultado é que a execução do AM ficou mais lenta por conta da implementação da Busca Local (BL), que não foi eficiente. O AM encontrou soluções com um número menor de gerações, porém seus resultados comparados com as do AG foram divergentes por conta da ineficiência da BL.

As Figuras 8, 9 e 10 ilustram, respectivamente, as melhores programações de plantio obtidas pelo AG para as 3 instâncias do PRC-A (10, 15 e 20 lotes). Nessas figuras, os números representam as culturas listadas na Tabela 2 (coluna Id.) e as cores iguais representam as culturas da mesma família botânica.

	Ano 1											
Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	1	8	8	6	6	6	6	30	25	25	25	1
2	16	30	24	24	24	7	7	12	12	12	16	16
3	1	8	8	6	6	6	6	30	25	25	25	1
4	16	30	24	24	24	7	7	12	12	12	16	16
5	1	8	8	6	6	6	6	30	25	25	25	1
6	16	16	30	24	24	24	7	7	12	12	12	16
7	1	8	8	6	6	6	6	30	25	25	25	1
8	23	23	23	7	7	18	18	18	18	18	30	23
9	1	8	8	6	6	6	6	30	25	25	25	1
10	16	16	30	24	24	24	7	7	12	12	12	16

Figura 8 - Melhor programação de plantio - AG - 10 lotes.

	Ano 1											
Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	1	8	8	6	6	6	6	30	25	25	25	1
2	16	16	30	24	24	24	7	7	12	12	12	16
3	1	8	8	6	6	6	6	30	25	25	25	1
4	16	16	30	24	24	24	7	7	12	12	12	16
5	1	8	8	6	6	6	6	30	25	25	25	1
6	16	30	24	24	24	7	7	12	12	12	16	16
7	1	8	8	6	6	6	6	30	25	25	25	1
8	16	16	30	24	24	24	7	7	12	12	12	16
9	1	8	8	6	6	6	6	30	25	25	25	1
10	23	23	23	7	7	18	18	18	18	18	30	23
11	16	16	30	24	24	24	7	7	12	12	12	16
12	1	8	8	6	6	6	6	30	25	25	25	1
13	16	30	24	24	24	7	7	12	12	12	16	16
14	1	8	8	6	6	6	6	30	25	25	25	1
15	16	16	30	24	24	24	7	7	12	12	12	16

Figura 9 - Melhor programação de plantio - AG - 15 lotes.

	Ano 1											
Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	16	16	30	24	24	24	7	7	12	12	12	16
2	1	8	8	6	6	6	6	30	25	25	25	1
3	16	16	30	24	24	24	7	7	12	12	12	16
4	1	8	8	6	6	6	6	30	25	25	25	1
5	16	16	30	24	24	24	7	7	12	12	12	16
6	1	8	8	6	6	6	6	30	25	25	25	1
7	16	16	30	24	24	24	7	7	12	12	12	16
8	23	23	23	7	7	18	18	18	18	18	30	23
9	16	16	30	24	24	24	7	7	12	12	12	16
10	1	8	8	6	6	6	6	30	25	25	25	1
11	1	8	8	6	6	6	6	30	25	25	25	1
12	16	16	30	24	24	24	7	7	12	12	12	16
13	1	8	8	6	6	6	6	30	25	25	25	1
14	23	23	23	7	7	1	1	12	12	12	30	23
15	18	30	30	24	24	24	7	7	18	18	18	18
16	16	16	30	24	24	24	7	7	12	12	12	16
17	1	8	8	6	6	6	6	30	25	25	25	1
18	16	16	30	24	24	24	7	7	12	12	12	16
19	1	8	8	6	6	6	6	30	25	25	25	1
20	23	23	23	7	7	1	1	12	12	12	30	23

Figura 10 - Melhor programação de plantio - AG - 20 lotes.

Para a instância com 10 lotes, o AG e o AM obtiveram a mesma solução. Assim, nas Figuras 11 e 12 são ilustradas, respectivamente, as melhores programações de plantio obtidas pelo AM para as outras 2 instâncias do PRC-A (15 e 20 lotes).

	Ano 1											
Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	1	8	8	6	6	6	6	30	25	25	25	1
2	16	16	30	24	24	24	7	7	12	12	12	16
3	23	23	7	7	6	6	6	6	30	30	23	23
4	16	16	30	24	24	24	7	7	12	12	12	16
5	23	23	23	7	7	6	6	6	6	30	30	23
6	16	16	30	24	24	24	7	7	12	12	12	16
7	23	23	7	7	30	6	6	6	6	30	23	23
8	1	30	30	24	24	24	7	7	12	12	12	1
9	23	23	7	7	6	6	6	6	30	30	23	23
10	18	18	18	30	7	7	30	25	25	25	18	18
11	16	16	30	24	24	24	7	7	12	12	12	16
12	1	8	8	6	6	6	6	30	25	25	25	1
13	16	30	24	24	24	7	7	12	12	12	16	16
14	23	23	7	7	6	6	6	6	30	30	23	23
15	16	16	30	24	24	24	7	7	12	12	12	16

Figura 11 - Melhor programação de plantio - AM - 15 lotes.

	Ano 1											
Lote/Mês	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	1	8	8	6	6	6	6	30	25	25	25	1
2	16	16	30	24	24	24	7	7	12	12	12	16
3	23	23	23	7	7	6	6	6	6	30	30	23
4	16	16	30	24	24	24	7	7	12	12	12	16
5	23	23	7	7	6	6	6	6	30	30	23	23
6	16	16	30	24	24	24	7	7	12	12	12	16
7	23	23	7	7	6	6	6	6	30	30	23	23
8	16	16	30	24	24	24	7	7	12	12	12	16
9	23	23	7	7	6	6	6	6	30	30	23	23
10	18	18	18	30	7	7	30	25	25	25	18	18
11	16	16	30	24	24	24	7	7	12	12	12	16
12	23	23	7	7	6	6	6	6	30	30	23	23
13	16	16	30	24	24	24	7	7	12	12	12	16
14	23	23	7	7	6	6	6	6	30	30	23	23
15	16	16	30	24	24	24	7	7	12	12	12	16
16	18	18	18	18	7	7	30	23	23	23	23	18
17	16	16	30	24	24	24	7	7	12	12	12	16
18	23	23	23	7	7	6	6	6	6	30	30	23
19	1	30	30	24	24	24	7	7	12	12	12	1
20	23	23	23	7	7	6	6	6	6	30	30	23

Figura 12 - Melhor programação de plantio - AM - 20 lotes.

Para melhor equiparar o comportamento dos algoritmos, nas Figuras 13, 14 e 15 são apresentados os gráficos com uma execução das 3 instâncias do PRC-A (10, 15 e 20 lotes). O eixo *x* indica as Gerações e o eixo *y* a Melhor Função Objetivo (MFO)

encontrada nas gerações dos algoritmos, ou seja, em cada geração do AG e do AM foi armazenado o valor da MFO.

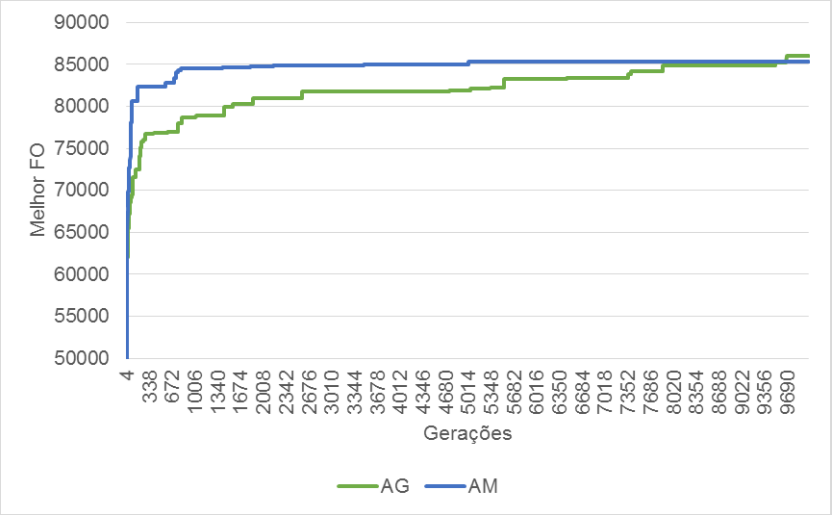

Figura 13 - Comportamento do AG e do AM (10 lotes) com restrição de tempo.

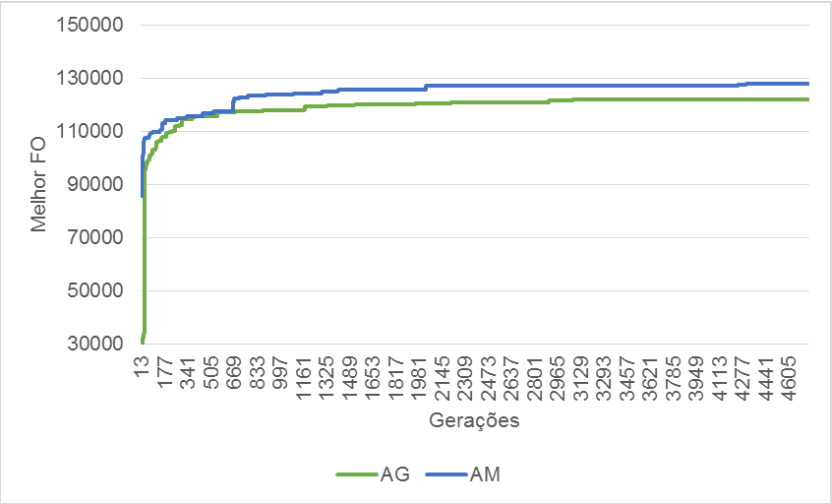

Figura 14 - Comportamento do AG e do AM (15 lotes) com restrição de tempo.

As 100 execuções realizadas com o AG e AM foram limitadas por um tempo de 5 minutos. As Figuras 13, 14 e 15 mostram que o AG e o AM obtiveram um comportamento esperado. O AG melhorou suas soluções gradativamente até estabilizar a solução, começando de um valor muito baixo e melhorando aos poucos. Já o AM consegue melhorar de forma mais significativa a MFO já nas primeiras gerações, isso devido à busca local.

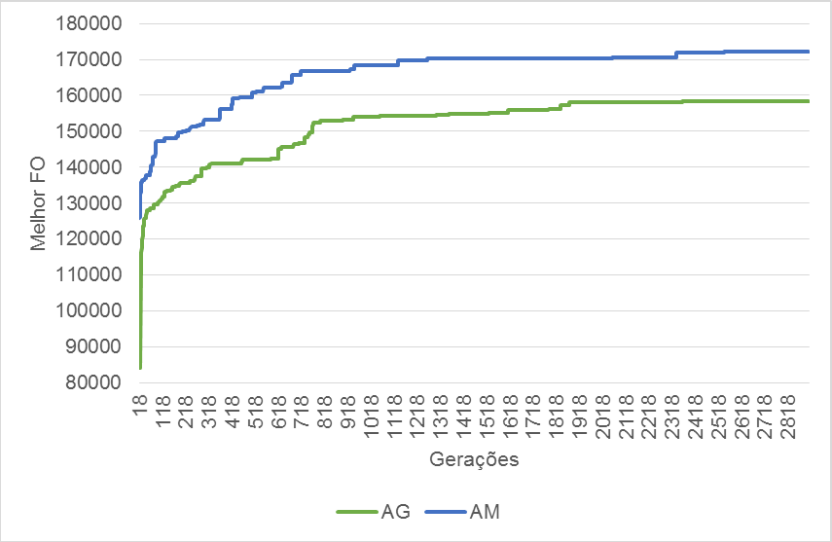

Figura 15 - Comportamento do AG e do AM (20 lotes) com restrição de tempo.

Para comparar a eficiência do AG e do AM, os algoritmos foram executados mais uma vez para as 3 instâncias, com os mesmos eixos (x para gerações e y para MFO). Agora, ambos foram executados com limitação de gerações, no máximo 10000, para equiparar os resultados. Os gráficos são apresentados nas Figuras 16, 17 e 18, respectivamente.

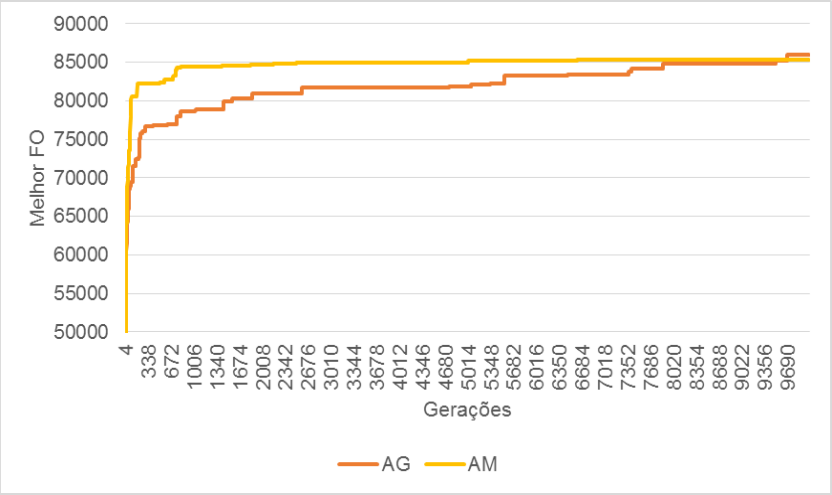

Figura 16 - Comportamentos do AG e do AM (10 lotes) em 10000 gerações.

O tempo gasto pelo AG para executar as 10000 gerações foi 20,66, 27,41 e 34,87 segundos, e o AM gastou 300,93, 643,16 e 1046,81 segundos para as instâncias com 10, 15 e 20 lotes, respectivamente. Em todos os gráficos (Figuras 13 a 18), o AM mostra que em poucas gerações o valor da MFO é aumentado em relação ao

AG. Porém, para a instância com 10 lotes o AG consegue encontrar uma solução melhor que o AM no final da execução.

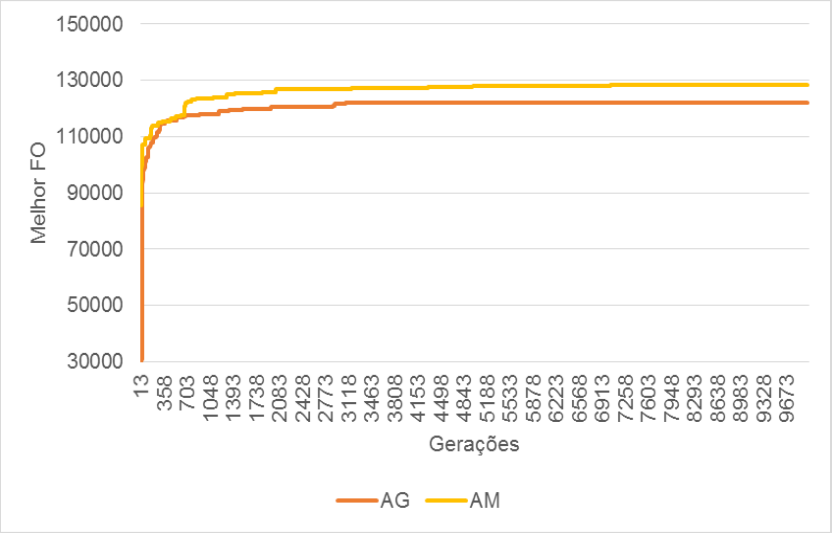

Figura 17 - Comportamentos do AG e do AM (15 lotes) em 10000 gerações.

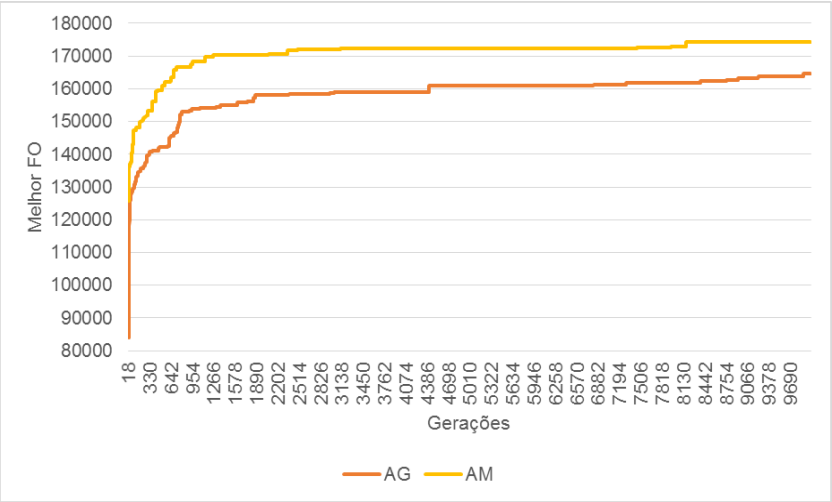

Figura 18 - Comportamentos do AG e do AM (20 lotes) em 10000 gerações.

4.2 COMPARAÇÃO COM A LITERATURA

As Tabelas 7 e 8 apresentam uma comparação entre a média das soluções obtidas pelo AG e pelo AM com os diversos métodos propostos por Aliano Filho (2012).

Observa-se na Tabela 7 que o AG desenvolvido neste trabalho apresentou uma melhora significativa em relação à média da lucratividade apresentada por Aliano Filho (2012), sendo 73,47% maior que o AG, 28,20% maior que o *Simulated Annealing* (SA), 46,33% maior que o AM e 29,46% maior que a heurística híbrida

AG+SA. O AM (Tabela 8) também apresentou melhora significativa em relação aos métodos apresentados por Aliano Filho (2012), sendo 70,93% maior que o AG, 26,33% maior que o SA, 44,19% maior que AM e 27,56% maior que o SA+AG.

Tabela 7 - Comparação das soluções obtidas pelo AG com a literatura.

Método	Instância			Total	Melhora (%)
	10 lotes	15 lotes	20 lotes		
AG	83347,50	127550,00	171637,50	382535,00	0
Métodos propostos por Aliano Filho (2012)					
AG	55022,00	74910,00	90590,00	220522,00	73,47
SA	68304,00	101200,00	128890,00	298394,00	28,20
AM	55369,00	86916,00	119140,00	261425,00	46,33
AG+SA	66985,00	91796,00	136708,00	295489,00	29,46

Tabela 8 - Comparação das soluções obtidas pelo AM com a literatura.

Método	Instância			Total	Melhora (%)
	10 lotes	15 lotes	20 lotes		
AM	83397,50	125265,00	168286,25	376948,75	0
Métodos propostos por Aliano Filho (2012)					
AG	55022,00	74910,00	90590,00	220522,00	70,93
SA	68304,00	101200,00	128890,00	298394,00	26,33
AM	55369,00	86916,00	119140,00	261425,00	44,19
AG+SA	66985,00	91796,00	136708,00	295489,00	27,56

Para as 100 execuções do AG e AM para as três instâncias do PRC-A, foi determinado um tempo de parada (5 minutos). Desse modo, foi utilizado para comparação o tempo médio para encontrar a melhor solução. Esse tempo, para cada método e instância, é apresentado na Tabela 9.

Tabela 9 - Comparação dos tempos computacionais médios obtidos.

Método	Instância			Tempo Médio (seg.)	Melhora (%)
	10 lotes	15 lotes	20 lotes		
AG	23,39	63,35	110,59	65,78	0
AM	177,03	250,95	276,23	234,74	-71,98
Métodos propostos por Aliano Filho (2012)					
AG	19,50	36,20	56,30	37,33	76,21
SA	49,34	104,26	181,16	111,58	-41,05
AM	4,13	9,71	22,62	12,15	441,40
AG+SA	6,38	10,65	13,67	10,23	543,01

Comparando os algoritmos propostos neste trabalho, o AG apresentou uma melhora de seu tempo em 71,98% em relação ao AM. Já comparando o AG desenvolvido com os métodos desenvolvidos por Aliano Filho (2012), obteve-se os seguintes

resultados: 76,21% pior que o AG, 41,05% melhor que o SA, 441,40% pior que o AM e 543,01% pior que AG+SA. Alguns resultados foram discrepantes, porém foram executados em máquinas e linguagens/aplicações diferentes.

Todos os experimentos deste trabalho foram realizados utilizando a linguagem de programação C++ em um computador com processador Intel Core I5 2.67GHz com 8GB de memória RAM. Já os experimentos computacionais realizados por Aliano Filho (2012) foram realizados com o *software* Matlab versão 7.4.0 R2007a em um computador com processador Core 2 Quad com 2GB de memória RAM.

5 CONCLUSÕES

Este trabalho propôs à aplicação de dois algoritmos evolutivos distintos para resolução de um Problema de Rotação de Culturas com Restrições de Adjacências (PRC-A). Foi implementado um Algoritmo Genético (AG) e, a partir deste, um Algoritmo Memético (AM).

Pôde-se observar que o PRC-A é um problema de Otimização Combinatória complexo, pois envolve diferentes culturas em diferentes situações e com um número considerável de restrições.

A partir dos resultados obtidos, o Algoritmo Genético (AG) se mostrou eficiente em relação às soluções obtidas por Aliano Filho (2012). Dentre todos os métodos apresentados, o AG obteve a melhor solução para as 3 instâncias do PRC-A.

Já o Algoritmo Memético (AM) também apresentou soluções melhores que as apresentadas por Aliano Filho (2012). Porém, essas soluções poderiam ser melhoradas com uma implementação mais eficiente e rápida da Busca Local (BL), que parece ter sido o “gargalo” do método.

Por fim, é possível concluir que os métodos propostos neste trabalho apresentam uma alternativa eficiente para auxiliar os produtores agrícolas, pois melhorou de forma significativa a lucratividade obedecendo às restrições para uma agricultura sustentável.

REFERÊNCIAS BIBLIOGRÁFICAS

ALIANO FILHO, A. *Metaheurísticas em um problema de rotação de culturas*. Dissertação (Mestrado em Biometria) – Universidade Estadual Paulista Júlio de Mesquita Filho (IBB–UNESP), Botucatu, 2012. Disponível em: <http://www.ibb.unesp.br/posgrad/teses/biometria_me_2012_angelo_filho.pdf>. Acesso em: 01 mai. 2013.

ALTIERI, M. *Agroecologia: bases científicas para uma agricultura sustentável*. 3 ed., Rio de Janeiro: Expressão Popular, 2012.

ARF, O.; SILVA, L. S.; BUZETTI, S. Efeito da rotação de culturas, adubação verde e nitrogenada sobre o rendimento do feijão. *Pesquisa Agropecuária Brasileira*, v.34, n.11, p. 2029-2036, 1999.

CHIPPERFIELD, A.; FLEMING, P. Multiobjective gas turbine engine controller design using genetic algorithms. *IEEE Transactions on Industrial Electronics*, v. 43, n. 5, p. 583-587, 1996.

COSTA, A. M.; SANTOS, L. M. R.; ALEM, D. J.; SANTOS, R. H. S. Sustainable vegetable crop supply problem with perishable stocks. *Annals of Operations Research*, doi: 10.1007/s10479-010-0830-y, 2011.

GLIESSMAN, S. R. *Agroecologia: processos ecológicos em agricultura sustentável*. 3 ed., Porto Alegre: Editora Universidade/UFRGS, 2005.

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

GOLDBERG, D. E. *Genetic algorithms in search, optimization and machine learning*. Boston: Addison-Wesley, 1989.

HOLLAND, J. H. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. Oxford, England: U Michigan Press, 1975.

MAURI, G. R.; ANDRADE, L. N.; LORENA, L. A. N. A memetic algorithm for a continuous case of the berth allocation problem. In. *International Conference on*

Evolutionary Computation Theory and Applications, 3, 2011, Paris – França. Proceedings… ECTA, 2011.

MOSCATO, P. Memetic algorithms: a short introduction. In. CORNE, D.; DORIGO, M.; GLOVER, F.; DASGUPTA, D.; MOSCATO, P.; POLI, R.; PRICE, K. V. (Ed.). *New ideas in optimization*. McGraw-Hill, p. 219-234, 1999.

SANTOS, L. M. R. *Programação de rotação de culturas: modelos e métodos de solução*. Tese (Doutorado em Ciências da Computação e Matemática Aplicada) – Universidade de São Paulo (ICMC–USP), São Carlos, 2009. Disponível em: <www.teses.usp.br/teses/disponiveis/55/55134/tde-09062009-110129/pt-br.php>. Acesso em: 17 de mai. 2013.

SANTOS, L. M. R.; COSTA, A. M.; ARENALES, M. N.; SANTOS, R. H. S. Sustainable vegetable crop supply problem. *European Journal of Operational Research*, v. 204, p. 639-647, 2010.

SANTOS, L. M. R.; MICHELON, P.; ARENALES, M. N.; SANTOS, R. H. S. Crop rotation scheduling with adjacency constraint. *Annals of Operations Research*, v. 190, n. 1, p. 165-180, 2011.

SANTOS, L. M. R.; SANTOS, R. H.; ARENALES, M. N.; RAGGI, L. A. Um modelo para a programação de rotações de culturas. *Pesquisa Operacional*, v.27, n. 3, p. 535-547, 2007.

SOUZA, M. J. F. *Notas de aula de inteligência computacional para otimização: Algoritmos Genéticos*. Universidade Federal de Ouro Preto (UFOP), 2012. Disponível em: <http://www.decom.ufop.br/prof/marcone/>. Acesso em: 16 ago. 2013.