

FLÁVIO LOPES RODRIGUES

**METAHEURÍSTICA E SISTEMA DE SUPORTE À DECISÃO NO
GERENCIAMENTO DE RECURSOS FLORESTAIS**

Tese apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência Florestal, para a obtenção do título de *Doctor Scientiae*.

BRASIL
VIÇOSA - MINAS GERAIS
2001

FLÁVIO LOPES RODRIGUES

METAHEURÍSTICA E SISTEMA DE SUPORTE À DECISÃO NO GERENCIAMENTO DE RECURSOS FLORESTAIS

Tese apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência Florestal, para a obtenção do título de *Doctor Scientiae*.

Aprovada: 02 de outubro de 2001.

Prof. Agostinho Lopes de Souza
(Conselheiro)

Prof. Heleno do Nascimento Santos
(Conselheiro)

Prof. Carlos Antônio Álvares
Soares Ribeiro

Prof. Márcio Lopes da Silva

Prof. Helio Garcia Leite

(Orientador)

*À força maior que guia e rege nossos
atos e sentimentos (Deus).
Aos meus familiares e amigos
pela amizade e confiança na triagem
desse longo caminho.
À minha noiva Mara.*

AGRADECIMENTOS

Manifesto minha sincera gratidão ao orientador e amigo, professor Helio Garcia Leite, pelo apoio, incentivo, compreensão e conhecimentos transmitidos durante o curso.

Aos Professores Agostinho Lopes de Souza, Heleno do Nascimento Santos, Carlos Antônio Álvares Soares Ribeiro e Márcio Lopes da Silva, pela valiosa colaboração nas sugestões apresentadas durante o desenvolvimento deste trabalho.

Aos amigos e brilhantes profissionais Joze Mauro e Eduardo Bering pela valorosa e indispensável contribuição no desenvolvimento desse trabalho. Aos amigos e colegas das longas jornadas: Gilciano, Valdir e Andreia, pela amizade e profissionalismo. Ao amigo Gilson, pela amizade, incentivos e lições de aprendizagem. Aos colegas e companheiros de república: Luís Carlos, Emerson, Vinícius, Ivan, Fernando, Gessionei, Jean e tantos outros, pela amizade, paciência e incentivos indispensáveis. Aos amigos de longa data Carlos Lopes, Tarcísio, Silvio Santos e Paulo Sérgio.

Aos meus familiares que sempre me ensinaram a resguardar e compreender os verdadeiros princípios e valores morais e espirituais, propiciando-me um verdadeiro crescimento pessoal e profissional, minha eterna gratidão. Em especial, agradeço à minha noiva Mara, pelo carinho e estima nos momentos decisivos.

A esta conceituada Instituição de ensino (UFV) e aos professores, funcionários e colegas do Departamento de Engenharia Florestal, pela oportunidade de conviver e adquirir novos e valiosos conhecimentos. Ao CNPq, pela concessão de bolsa de estudos.

A todos que aqui citei e aos que injustamente não foram mencionados, mas que contribuíram de alguma forma para a realização desse trabalho, o meu muito obrigado.

BIOGRAFIA

FLÁVIO LOPES RODRIGUES, filho de Francisco de Assis Rodrigues e Mercês de Souza Lopes, nasceu na cidade de São Miguel do Anta, Estado de Minas Gerais.

Em 1990 iniciou-se no Curso de Engenharia Florestal, na Universidade Federal de Viçosa - MG, graduando em agosto de 1994.

Durante a graduação, entre as diversas atividades em que participou, destaca-se o desenvolvimento de pesquisas básicas durante três anos e meio como bolsista de iniciação científica pelo CNPq.

Em Agosto de 1994, iniciou o curso de Pós-graduação em Ciência Florestal, pela Universidade Federal de Viçosa - MG, tendo defendido tese em fevereiro de 1997.

Em setembro de 1997, iniciou o curso de Doutorado em Ciência Florestal, na Universidade Federal de Viçosa, defendendo tese em 02 de outubro de 2001.

ÍNDICE

Página

RESUMO	XII
ABSTRACT.....	XIV
1. INTRODUÇÃO	XVI

CAPÍTULO 1	XIX
-------------------------	------------

TOMADA DE DECISÃO E O GERENCIAMENTO DE RECURSOS FLORESTAIS: MODELOS, DADOS, TECNOLOGIAS E RECURSOS HUMANOS	XIX
---	------------

1. INTRODUÇÃO	XXI
2. TOMADA DE DECISÃO E O PROCESSO DE GERÊNCIA	XXIII
3. PROBLEMAS DE MANEJO FLORESTAL	XXV
4. DADOS E INFORMAÇÕES	XXXI
5. SISTEMAS DE SUPORTE À DECISÃO.....	XXXV
5.1. Modelos utilizados em SSD florestal	xi
5.1.1. Modelos de prognose da produção.....	xii
5.1.2. Modelos econômicos.....	xliv
5.1.3. Modelos de regulação florestal.....	xlv
5.2. Alguns equívocos no uso de modelos	li
5.3. Tecnologia e o processo decisório	liii
6. CONSIDERAÇÕES FINAIS	LVII
7. REFERÊNCIAS BIBLIOGRÁFICAS	LVIII

CAPÍTULO 2	LXIII
-------------------------	--------------

SISFLOR: UM SISTEMA DE SUPORTE À DECISÃO PARA O PLANEJAMENTO DE FLORESTAS EQUIÂNEAS	LXIII
--	--------------

1. INTRODUÇÃO	LXV
2. MATERIAIS E MÉTODOS	LXVII
2.1. Desenvolvimento do sistema	lxvii
2.2. Estudo de caso	lxvii
2.2.1. Definição do problema	lxvii
2.2.2. Unidades de manejo	lxviii
2.2.3. Estimativa da produção	lxix
2.2.4. Custos das operações florestais	lxix
2.2.5. Formulação do problema.....	lxx
2.2.5.1. Programação linear	lxx
2.2.5.2. Programação inteira.....	lxxiii
3. RESULTADOS E DISCUSSÃO.....	LXXV
3.1. O Sistema de suporte à decisão	lxxv
3.1.1. Módulo para entrada de dados e informações	lxxvi

3.1.1.1. Unidades de manejo	lxxvi
3.1.1.2. Regime de manejo	lxxvii
3.1.1.3. Material genético	lxxviii
3.1.1.4. Prognóstico	lxxix
3.1.1.5. Custos.....	lxxix
3.1.1.6. Destinos da madeira	lxxx
3.1.1.7. Usos da madeira.....	lxxx
3.1.1.7. Preços de venda de terras	lxxxii
2.1.1.8. Preços da madeira.....	lxxxii
3.1.1.9. Outros módulos.....	lxxxii
3.1.2. Módulo para modelagem.....	lxxxii
3.1.2.1. A matriz de manejo	lxxxiii
3.1.2.2. Substituição de material genético	lxxxix
3.1.2.3. Modelos de silvicultura.....	xc
3.1.2.4. Gerador de alternativas de manejo	xcii
3.1.2.5. Restrições do modelo	xcii
3.1.3. Módulo gerador de modelos	xcv
3.1.4. Módulo de otimização	xcvi
3.1.5. Módulo de relatórios.....	xcvi
3.2. Solução do problema proposto	xcvii
4. CONCLUSÕES	CII
5. REFERÊNCIAS BIBLIOGRÁFICAS	CIII

CAPÍTULO 3CV

PRINCÍPIOS E APLICAÇÕES DE ABORDAGENS HEURÍSTICAS PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTALCV

1. INTRODUÇÃO	CVIII
2. HEURÍSTICAS	CIX
3. ALGORITMOS GENÉTICOS	CXIII
3.1. Princípios básicos	cxiv
3.2. Elementos do algoritmo genético	cxv
3.2.1. Codificação das soluções do problema em termos de cromossomas	cxvi
3.2.2. População inicial	cxvii
3.2.3. Seleção	cxviii
3.2.4. Implementação da seleção	cxxiii
3.2.5. Operadores genéticos	cxxiii
3.2.6. Parâmetros de controle de um algoritmo genético	cxxv
3.2.7. Formalização do algoritmo genético	cxxvi
4. BUSCA TABU	CXXVII
4.1. Filosofia do método	cxxvii
4.2. Princípios básicos	cxxviii
4.3. Vizinhanças e movimentos.....	cxxx
4.4. Memória e classificação tabu.....	cxxxi
4.4.1. Memória de curto prazo	cxxxii

4.4.1.1. Permanência tabu.....	cxxxiii
4.4.2. Memória de longo prazo	cxxxv
4.4.2.1. Abordagem baseada na frequência.....	cxxxv
4.4.2.2. Estratégias de intensificação	cxxxvi
4.4.2.3. Estratégias de diversificação	cxxxvi
4.5. Lista tabu.....	cxxxvii
4.6. Atributos de movimento	cxxxviii
4.7. Critérios de aspiração	cxxxviii
4.8. Oscilação estratégica	cxxxix
4.9. Classificação da busca tabu.....	cxl
5. <i>SIMULATED ANNEALING</i>	CXLI
5. MÉTODOS METAHEURÍSTICOS HÍBRIDOS.....	CXLV
6. MEDIDAS DE PERFORMANCE.....	CXLVI
7. APLICAÇÕES NO MANEJO FLORESTAL.....	CXLVII
8. CONCLUSÕES.....	CXLIX
9. REFERÊNCIAS BIBLIOGRÁFICAS	CL

CAPÍTULO 4	CLIV
-------------------------	------

METAHEURÍSTICA ALGORITMO GENÉTICO PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL COM RESTRIÇÕES DE INTEIREZA	CLIV
--	------

1. INTRODUÇÃO	CLV
2. MATERIAIS E MÉTODOS	CLVIII
2.1. Descrição dos problemas	clviii
2.2. Formulação dos problemas	clviii
2.3. Desenvolvimento do AG	clxii
2.3.1. Codificação computacional do AG.....	clxii
2.3.2. Entrada de dados	clxiii
2.3.3. Codificação dos indivíduos.....	clxiii
2.3.4. Função de avaliação	clxv
2.3.4.1. Método das penalidades.....	clxvi
2.3.5. Geração da população inicial.....	clxvi
2.3.6. Seleção dos indivíduos	clxvii
2.3.7. Implementação dos métodos de seleção.....	clxx
2.3.8. Operadores genéticos.....	clxxi
2.3.8.1. Operador de <i>crossover</i>	clxxi
2.3.8.1.1. <i>Crossover</i> de um único ponto	clxxii
2.3.8.1.2. <i>Crossover</i> de dois pontos	clxxiii
2.3.8.1.3. <i>Crossover</i> multipontos	clxxiv
2.3.8.2. Operador de mutação	clxxv
2.3.9. Critério de parada.....	clxxvi
2.3.10. Escolha dos parâmetros do AG.....	clxxvii
2.3.11. Avaliação do desempenho do AG.....	clxxx
2.3.12. Análise estatística dos resultados.....	clxxxi

3. RESULTADOS E DISCUSSÃO.....	CLXXXI
3.1. O plano de manejo	clxxxi
3.2. Ilustração de uma corrida do AG.....	clxxxiii
3.3. Avaliação dos parâmetros e medidas de performance do AGcxc
4. CONCLUSÕES.....	CXCV
5. REFERÊNCIAS BIBLIOGRÁFICAS	CXCVII

CAPÍTULO 5 CXCVIII

METAHEURÍSTICA BUSCA TABU PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL COM RESTRIÇÕES DE INTEIREZA

..... CXCVIII

1. INTRODUÇÃO	CXCIX
2. MATERIAL E MÉTODOS.....	CC
2.1. O Problema de planejamento florestal.....	cc
2.2. O Algoritmo BT.....	cci
2.2.1. Entrada de dados	ccii
2.2.2. Codificação da solução	ccii
2.2.3. Solução inicial	cciii
2.2.4. Vizinhança e movimentos	cciii
2.2.4.1. Vizinhança total.....	cciii
2.2.4.2. Vizinhança parcial.....	ccv
2.2.5. Função de avaliação	ccv
2.2.6. Memória de curto prazo e restrições tabu.....	ccvi
2.2.7. Lista tabu e atributos do movimento	ccviii
2.2.8. Critérios de aspiração	ccx
2.2.9. Critério de parada.....	ccx
2.3. Escolha dos parâmetros da BT.....	ccxi
2.4. Avaliação da performance da BT	ccxii
2.5. Análise estatística dos resultados.....	ccxiii
3. RESULTADOS E DISCUSSÃO.....	CCXIII
3.1. Ilustração de uma corrida da BT	ccxiii
3.2. Avaliação dos parâmetros e medidas de performance da BT	ccxxii
4. CONCLUSÕES.....	CCXXVIII
5. REFERÊNCIAS BIBLIOGRÁFICAS	CCXXIX

CAPÍTULO 6 CCXXXI

METAHEURÍSTICA *SIMULATED ANNEALING* PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL COM RESTRIÇÕES DE INTEIREZA

..... CCXXXI

1. INTRODUÇÃO	CCXXXIII
2. MATERIAIS E MÉTODOS.....	CCXXXIV
2.1. O problema de planejamento florestal	ccxxxiv

2.2. O Algoritmo <i>Simulated Annealing</i>	ccxxxiv
2.2.1. Entrada de dados	ccxxxv
2.2.2. Codificação da solução	ccxxxv
2.2.3. Solução inicial	ccxxxvi
2.2.4. Função de avaliação	ccxxxvi
2.2.5. Movimentos	ccxxxvii
2.2.6. Temperatura	ccxxxix
2.2.7. Critério de parada.....	ccxl
2.3. Escolha dos parâmetros da SA.....	ccxl
2.4. Avaliação do desempenho da SA.....	ccxlii
2.5. Análise estatística dos resultados.....	ccxlii
3. RESULTADOS E DISCUSSÃO.....	CCXLII
3.1 Ilustração de uma corrida da heurística SA.....	ccxliii
3.2. Avaliação dos parâmetros e medidas de performance da SA	ccxlv
4. CONCLUSÕES.....	CCL
5. REFERÊNCIAS BIBLIOGRÁFICAS	CCLI
RESUMO E CONCLUSÕES	CCLIII

RESUMO

RODRIGUES, Flávio Lopes, D.S., Universidade Federal de Viçosa, outubro de 2001.
Metaheurística e Sistema de Suporte à Decisão no Gerenciamento de Recursos Florestais. Orientador: Helio Garcia Leite. Conselheiros: Agostinho Lopes de Souza e Geraldo Galdino de Paula Junior.

Os objetivos gerais deste trabalho foram: avaliar a contribuição dos modelos, dados, tecnologias e recursos humanos no processo de tomada de decisão para o gerenciamento florestal; desenvolver e apresentar um sistema de suporte à decisão com novas abordagens para o gerenciamento de florestas equiâneas; e desenvolver e testar as metaheurísticas Algoritmo Genético (AG), Busca Tabu (BT) e *Simulated Annealing* (SA) para a solução de problemas de planejamento florestal com restrições de inteireza. Para alcançar seus objetivos, este estudo foi dividido em seis capítulos. No capítulo 1, fez-se uma revisão sobre o uso de modelos, dados, tecnologias e recursos humanos para tomada de decisão no gerenciamento florestal. O estudo revela a necessidade de se considerar a integração desses recursos, apontando ainda as principais falhas e equívocos nos seus usos pelas empresas florestais. No capítulo 2 é apresentado o SisFlor, um Sistema de Suporte à Decisão (SSD) flexível e amigável, que permite formular e resolver, através de modelos de Programação Linear (PL) e Programação Inteira (PI) importantes questões do gerenciamento de florestas equiâneas. As limitações do algoritmo exato *branch and bound* para solução dos modelos de PI, formulados através do SSD SisFlor é a principal motivação para desenvolvimento dos capítulos seguintes. No capítulo 3 é apresentada uma revisão sobre as heurísticas, as mais promissoras abordagens para solução de problemas de PI. Dentre as abordagens heurísticas, foram selecionadas as metaheurísticas AG, BT e SA, discutidas à luz dos seus princípios básicos e de algumas aplicações no gerenciamento florestal. Nos capítulos 4, 5 e 6 foram formuladas e implementadas em um código computacional as metaheurísticas AG, BT e SA. Para testar estas três metaheurísticas

foram selecionados cinco problemas, contendo entre 12 e 423 variáveis de decisão, com restrições de singularidade, produção mínima e produção máxima. Todos os problemas tiveram como objetivo a maximização do valor presente líquido. O problema um foi utilizado para ilustração de uma corrida das respectivas metaheurísticas. Os demais problemas foram utilizados para avaliar os efeitos de diversos parâmetros dessas metaheurísticas, avaliados segundo medida de eficácia, dada pela relação entre o melhor valor da função objetivo obtida pela metaheurística e o ótimo matemático obtido pelo algoritmo exato *branch and bound*. Os parâmetros das metaheurísticas avaliadas foram comparados através do teste L&O e a análise por estatísticas descritivas. Os melhores valores dos parâmetros selecionados para as respectivas metaheurísticas forneceram eficácia média de 95,97% para a BT, 95,36% para a SA e 94,28% para o AG. Em relação aos valores máximos de eficácia obtidos, SA foi a metaheurística que apresentou maior valor para um dos problemas (100%), seguida pela BT (98,84%) e AG (98,48%). O maior coeficiente de variação foi obtido pela SA (3,18%), seguida pela BT (2,48%) e AG (2,08%). A eficiência, medida pelo tempo de processamento para obtenção da solução, obtida pela BT, AG e SA foram cerca de duas, cinco e dez vezes superior ao algoritmo *branch and bound*, respectivamente. As três metaheurísticas apresentaram-se como abordagens atrativas para solução de problemas combinatoriais importantes no contexto do gerenciamento florestal, problemas esses, de difícil solução ou até mesmo impossíveis de serem solucionados pelos algoritmos exatos da atualidade.

Palavras-chave: Gerenciamento Florestal, Sistema de Suporte à Decisão, Pesquisa Operacional, Heurísticas.

ABSTRACT

RODRIGUES, Flávio Lopes, D.S., Universidade Federal de Viçosa, October, 2001.
Metaheuristic and Decision Support System in Forest Resource Management. Adviser: Helio Garcia Leite. Committee Members: Agostinho Lopes de Souza and Geraldo Galdino de Paula Junior.

The objectives of this work were: to evaluate the contribution of models, data, technologies and human resources in the process of decision making for forest management; to develop and present a decision support system with new approaches to even-age forest management; and to develop and test the metaheuristic Genetic Algorithm (GA), Tabu Search (TS) and Simulated Annealing (SA) to solve problems of forest planning with integer constraints. To accomplish these objectives, this study was divided in six chapters. Chapter 1 comprises a review on the use of models, data, technologies and human resources to make decision in forest management. The study reveals the need of considering the integration of those resources, showing still the major drawbacks and misunderstandings in their use by forest companies. Chapter 2 presents SisFlor, which is a flexible Decision Support System (DSS) with a friendly interface to the user that allows to formulate and solve via Linear Programming (LP) and Integer Programming (IP) models important issues of even-age forest management. The limitations of the exact algorithm *branch and bound* to solve IP models, formulated through SisFlor DSS, is the main reason to develop the following chapters. Chapter 3 presents a review on the heuristics, the most promising approaches for solving IP problems. Among the heuristic approaches, the metaheuristic GA, TS and SA were selected and discussed on the basis of their basic principles and of some applications to forest management. In chapters 4, 5 and 6 the metaheuristic GA, TS and SA were formulated and implemented into a computational code. To test these metaheuristics, five problems were selected containing between 12 and 423 decision variables, with constraints of singularity and minimum and maximum production. All the problems had the objective of maximizing the net present value. Problem one was used for illustrating a race of the concerning metaheuristics. The other problems were

used to evaluate the effects of several parameters of those metaheuristics, that were evaluated according to a efficacy measure given by the relation between the best value of the objective function obtained by the metaheuristic and the mathematical optimum obtained by the exact algorithm *branch and bound*. The parameters of the evaluated metaheuristics were compared by the L&O test and the analysis by descriptive statistics. The best parameter values selected for the concerning metaheuristics gave average efficacy of 95.97% for TS, 95.36% for SA and 94.28% for GA. In relation to the maximum efficacy values obtained, SA presented larger value for one of the problems (100%), followed by TS (98.84%) and GA (98.48%). The largest coefficient of variation was obtained by SA (3.18%), followed by TS (2.48%) and AG (2.08%). The efficiency, which was measured by the processing time for obtaining the solution through TS, GA and SA, was about two, five and ten times superior to the algorithm *branch and bound* respectively. Those metaheuristics were shown as attractive approaches for the solution of important combinatorial problems within the forest management context, problems of difficult or even impossible solution by the current exact algorithms.

Key-words: Forest Management, Decision Support System, Operational Research and Heuristic.

1. INTRODUÇÃO

A obtenção de informações com qualidade e rapidez é fundamental para o sucesso das tomadas de decisão em manejo florestal, sendo dependente do uso de métodos e técnicas da ciência gerencial, que permitem avaliar situações alternativas, analisar os resultados e reduzir riscos e incertezas na tomada de decisão. A utilização dessas técnicas propicia melhoria na alocação de recursos produtivos, aumentando a rentabilidade econômica das empresas (SILVA Jr, 1993). Dentre as diversas técnicas gerenciais cabe destacar os modelos de Pesquisa Operacional (PO), surgidos durante a segunda guerra mundial. Atualmente pode-se encontrar aplicações da PO em quase todas as áreas da atividade humana (HILLIER e LIEBERMAN, 1988).

Entre as técnicas de PO com aplicações no manejo de recursos florestais, estão: a programação linear (PL), programação inteira (PI), programação com múltiplos objetivos (PMO), programação dinâmica (PD) e simulação.

A PL tem sido utilizada especialmente em problemas de ordenamento da produção florestal (CURTIS, 1962; LOUCKS, 1964; NAUTYAL & PEARSE, 1967; NAVON, 1971; WARE e CLUTTER, 1971). Apesar das enormes potencialidades, algumas de suas limitações tem levado à necessidade de novos estudos e de modelos específicos de suporte à decisão. Por exemplo, as limitações da PL para solução de problemas envolvendo múltiplos objetivos foram equacionadas através da utilização dos modelos de PMO (FIELD, 1973; BARE e MENDOZA, 1988). O sucesso da PL e PMO deve-se à alta eficiência do método *simplex*, um dos principais algoritmos de solução desses modelos, disponível em diversos pacotes comerciais. As preocupações com o uso mais adequado dos recursos florestas têm exigido novas abordagens para solução de alguns problemas clássicos de regulação florestal. A necessidade de manter a integridade espacial de unidades de manejo ao longo do tempo e de impor restrições de corte em talhões adjacentes são alguns exemplos de situações que têm resultado em

novas abordagens para solução de problemas de regulação florestal. Tais restrições podem ser formuladas através de modelos de PI, conforme encontrado em TORRES-ROJO e BRODIE (1990), YOSHIMOTO e BRODIE (1993) e MURRAY (1999). Entretanto, a formulação de modelos matemáticos de PI para análise de problemas de planejamento florestal, podem resultar em modelos de natureza combinatorial, algumas vezes de difícil solução. Segundo GOLDBARG e LUNA (2000), o cerne da dificuldade da abordagem exata dos problemas denominados NP-árduos, onde se encontram grande parte dos problemas de PI, está na explosão combinatória dos métodos enumerativos. Algoritmos como *branch and bound*, *implicit enumeration* e *cutting plane*, contudo, são aplicáveis apenas em problemas de tamanho moderado (IGNIZIO e CAVALLER, 1994), o que certamente não é o caso das aplicações florestais, que em geral envolvem centenas ou milhares variáveis de decisão. Alguns autores tem sugerido um conjunto de técnicas e algoritmos computacionais muito eficientes, embora não garantam a solução ótima para os problema de PI. Esses algoritmos são denominados de *heurísticos* ou *aproximativos* (GOLDBARG e LUNA, 2000).

Independente da técnica utilizada, desenvolver e utilizar Sistemas de Suporte à Decisão (SSD) tem se tornado cada vez mais imprescindível. Apesar disso, no Brasil, o uso de SSD é ainda incipiente, justificando a continuidade da pesquisa sobre os mesmos. Assim, conduziu-se este estudo visando contribuir para a implementação de importantes ferramentas de suporte à decisão. No primeiro capítulo é apresentado uma reflexão sobre papel dos modelos, dados, tecnologias e recursos humanos em processos de tomada de decisão, com enfoque no manejo florestal. No segundo é apresentado o SisFlor, um sistema de suporte à decisão em manejo florestal. Nesse mesmo capítulo são discutidas algumas limitações de aplicações em alguns problemas de planejamento florestal dos modelos de PL e PI, por meio de um estudo de caso. No terceiro capítulo discute-se os princípios básicos de três técnicas heurísticas: Algoritmo Genético, Busca Tabu e *Simulated Annealing*, destacando as potencialidades dessas técnicas como alternativa para solução de problemas de planejamento florestal com restrições de inteireza. No quarto, quinto e sexto capítulo

são apresentadas algumas aplicações das heurísticas Algoritmo Genético, Busca Tabu e *Simulated Annealing* por meio de estudos de casos.

CAPÍTULO 1

TOMADA DE DECISÃO E O GERENCIAMENTO DE RECURSOS FLORESTAIS: MODELOS, DADOS, TECNOLOGIAS E RECURSOS HUMANOS

RESUMO - Este trabalho teve como objetivo avaliar a contribuição dos modelos, dados, tecnologias e recursos humanos para tomada de decisão no gerenciamento de recursos florestais. Através de uma ampla revisão de literatura, o processo de tomada de decisão foi discutido, considerando-se as suas diferentes fases, as características dos problemas enfrentados pelos tomadores de decisão e a necessidade de se considerar o uso de diferentes modelos, dados e tecnologias nesse processo. Dentre os diferentes modelos utilizados no processo de tomada de decisão no gerenciamento florestal, foi dada uma atenção especial aos econômicos, de pesquisa operacional e de prognose da produção florestal. A necessidade de dados com quantidade e qualidade para alimentação desses modelos foi também ressaltada como fator determinante para produção de informações com qualidade. A importância da tecnologia foi ressaltada pela necessidade e importância dos sistemas de informações, com enfoque especial no uso dos sistemas de suporte à decisão para o gerenciamento florestal. O papel e a importância dos recursos humanos foram também avaliados como peça fundamental do processo decisório. Finalmente, fez-se uma reflexão sobre a necessidade de integração desses recursos, apontando as principais falhas e equívocos em seus usos para auxiliar o processo de tomada de decisão nas empresas florestais.

Palavras-chave: Gerenciamento florestal, modelos, dados, tecnologias, recursos humanos

DECISION MAKING AND FOREST RESOURCE MANAGEMENT: MODELS, DATA, TECHNOLOGIES AND HUMAN RESOURCES

ABSTRACT - The objective of this work was to evaluate the contribution of models, data, technologies and human resources for decision making in forest resource management. Through a wide literature review, the process of decision making was discussed, considering its different phases, characteristics of the problems faced by the decision makers, and the need of considering the use of different models, data and technologies in that process. Among the different models used in the process of decision making in forest management, an economic special attention was given, of operational research on forest production prognosis. The need for data in quantity and quality to feed those models was also pointed out as a decisive factor for the production of quality information. The importance of technology was pointed out by the need and importance of information systems, with special focus on the use of decision support systems toward forest management. The role and the importance of human resources were also appraised as a fundamental part of the decision process. Finally, the need for integration of those resources was considered, pointing out the main drawbacks and misunderstandings in their uses to assist the decision making process in forest companies.

Key-words: Forest Management, Operational Research, Heuristic, Data, Models, Technology and Resources Human .

1. INTRODUÇÃO

A solução de problemas de planejamento florestal envolve a identificação e definição do problema, a coleta, processamentos e armazenamento de dados, a geração de alternativas, a escolha de modelos e de tecnologias, o monitoramento e controle, e a capacidade dos indivíduos em lidar com esses fatores (TURBAN e ARONSON, 1998). Uma abordagem usada para análise e solução de problemas de planejamento florestal é a pesquisa operacional (PO), em especial a programação linear (PL). A eficiência do algoritmo *Simplex*, desenvolvido por Dantzig em 1947, utilizado para solução de problemas de PL de grande porte impulsionou o uso desses modelos no setor. Entretanto, a incapacidade de solução de problemas com múltiplos objetivos conflitantes e restrições de inteireza tem limitado o uso dos mesmos a poucos problemas reais. Surgiram, então, os modelos de programação com múltiplos objetivos (PMO) e de programação inteira (PI). Enquanto os objetivos conflitantes foram adequadamente tratados pela PMO e solucionados eficientemente pelo método *Simplex*, as restrições de inteireza demandaram formulações de PI. A incorporação dessas restrições nos modelos de planejamento, incorporou também grandes dificuldades no seu processo de solução. Problemas tratados como lineares contínuos (PL e PMO) foram transformados em problemas lineares discretos (PI), de natureza combinatorial, resolvidos, em geral, através de algoritmos exatos como *branch and bound*, *cutting planning e implicit enumeration*, restritos à problemas de pequeno porte (IGNIZIO e CAVALLER, 1994).

A solução dos modelos de PI ganhou novo enfoque com o advento das técnicas heurísticas (IGNIZIO e CAVALIER (1994), que embora não resultem em garantia de otimalidade das soluções, podem oferecer soluções aceitáveis, inclusive para problemas complexos e de grande porte, com baixo custo computacional.

Apesar dos grandes avanços na pesquisa e da utilização de Sistemas de Suporte à Decisão (SSD) em diversos setores da sociedade, em manejo de recursos florestais, o uso de SSD é ainda incipiente. Em parte, isto é devido ao desconhecimento, por parte de alguns indivíduos, da potencialidade dos métodos de PO e da carência de interfaces eficientes e de baixo custo, para solução de diferentes problemas de manejo florestal. A utilização de sistemas computadorizados é indispensável para viabilizar o processamento e armazenamento do volume de dados, gerar novas informações, manipular de forma mais fácil os modelos utilizados em diferentes situações, pesquisar as soluções do problema e oferecer recursos para apresentação de resultados. Contudo desenvolver tais sistemas não é uma tarefa fácil de executar. É necessário conhecer classes de problemas, dados, tecnologias e recursos humanos envolvidos nos problemas de decisão. A multidisciplinaridade é, portanto, um aspecto indispensável na confecção de SSD. Conhecimentos de técnicas de gerenciamento e análise de sistemas é fundamental na confecção de um SSD. No gerenciamento florestal, o papel do engenheiro florestal é uma peça chave nesse processo, uma vez que, com formação em diferentes áreas da ciência, propicia tratamentos científicos adequados à formulação e solução dos problemas de gerenciamento de recursos florestais. Além disso, o sucesso da análise, interpretação e implementação dos resultados obtidos pelos planos formulados através dos SSD dependem da competência desses profissionais. Os

objetivos específicos deste estudo foram:

- discutir as principais características dos problemas enfrentados pelos manejadores florestais;
- apresentar o método científico para tomada de decisão, enfocando o papel dos dados, modelos, tecnologias e recursos humanos nesse processo;
- apresentar uma revisão da evolução dos modelos e tecnologias utilizados no manejo de recursos florestais;
- discutir sobre os principais problemas e equívocos apontados em diversas experiências quando da utilização dos modelos de gerenciamento de recursos florestais;

- discutir sobre o papel dos sistemas de informação no planejamento florestal, apontando também os problemas e equívocos mais comuns nesses sistemas.

2. Tomada de decisão e o processo de gerência

Gerenciar pode ser definido como o esforço para alcançar metas por meio de planejamento, organização, liderança e controle. Nesse conceito, planejamento refere-se à definição de metas e como estas serão atingidas. Organizar consiste em estabelecer responsabilidades para realização de tarefas e indicar os recursos apropriados. Liderança significa motivar empregados para alcançarem as metas organizacionais. O controle envolve o monitoramento de atividades e correções quando necessário. Além de planejar, organizar, liderar e controlar, os gerentes devem estimular, cuidar, informar, motivar e decidir (LAUDON e LAUDON, 1999). O planejamento tem um papel decisivo no processo de gerência, uma vez que os planos são as bases por meio das quais as organizações obtêm e aplicam recursos para alcançar seus objetivos, permitindo medir e monitorar o alcance dos objetivos e adotar medidas corretivas, quando necessário (STOONER e FREEMAN, 1985). Neste sentido, MEGGINSON et al. (1986) ampliaram as funções do planejamento ao relatarem que planejar refere-se a escolher um curso de ação e decidir antecipadamente o que deve ser feito, em que seqüência, quando e como, proporcionando bases para ações administrativas efetivas. Segundo esses autores, o planejamento é importante para: i) ajudar a administração face às mudanças do ambiente; ii) permitir que os administradores vejam o quadro operativo interno com maior clareza; iii) ajudar no estabelecimento de responsabilidade; iv) proporcionar ordem às operações e coordenação entre as várias partes da organização; v) fazer com que os objetivos sejam mais específicos e bem conhecidos; vi) minimizar a adivinhação; e poupar tempo, esforço e dinheiro.

Em função dos níveis de abordagens e objetivos, o planejamento pode ser classificado em: estratégico, tático e operacional (TAVARES, 1991). O primeiro abrange toda a organização e tem características principais como: horizonte de planejamento de longo prazo, é definido pela cúpula da organização,

correspondendo ao plano maior, ao qual todos os demais estão subordinados. O Segundo é executado em nível de departamento, sendo projetado para horizontes de médio prazo, geralmente anual. O planejamento operacional é mais detalhado, sendo executado para cada tarefa ou atividade, projetado para horizonte de planejamento de curto prazo e é destinado ao alcance das metas específicas.

É evidente que para um gerente cumprir o seu papel, contribuindo para o atendimento dos objetivos, é necessário um contínuo processo de tomada de decisão (atividade inerente ao planejamento), que exige bons julgamentos, experiência, análise quantitativa, criatividade, suposições e conhecimento do ambiente envolvido (STOONER e FREEMAN, 1985).

O processo de tomada de decisão é constituído por quatro fases (TURBAN e ARONSON, 1998): inteligência, projeto, escolha e implementação (Figura 1).

O processo de Tomada de Decisão (TD) inicia-se com o exame da realidade, onde um possível problema pode ser identificado e definido (fase inteligente). Um problema surge quando um indivíduo não está satisfeito com a realidade presente ou prevista e sua identificação e definição ocorre quando se busca modificar a realidade e alcançar a situação futura desejada (TURBAN e ARONSON, 1998). A coleta de dados e informações é uma das principais atividades dessa etapa sendo decisivos na formulação do problema.

Uma vez identificado e definido um problema, é necessário escolher ou construir um modelo que o represente. Esta é a fase de projeto, na qual o modelo, construído através de simplificações e suposições, é utilizado gerando subsídios para tomada de decisão. O modelo mais apropriado para um dado problema depende da natureza matemática das relações entre as variáveis envolvidas, dos objetivos, da extensão e do controle sobre as variáveis e do nível de incerteza associado (ANDRADE, 1990). Construir o modelo requer a pesquisa e geração de alternativas de ação, em geral utilizando-se rotinas computacionais (Hash *et al.*, 1981), citados por SILVA JUNIOR, 1993). A fase de projeto termina com a validação e verificação do modelo, podendo levar a necessidade de coleta de novos dados e redefinição do problema (TURBAN e ARONSON, 1998).

O próximo passo é a escolha de uma ou mais alternativas geradas na fase de projeto, ou seja, a fase de escolha, onde são adotadas ferramentas de suporte à decisão, por exemplo, modelos de pesquisa operacional e matemática financeira.

Após teste de adequação da solução do modelo tem-se a última etapa do processo (fase de implementação da solução) que culmina com a verdadeira solução do problema (TURBAN e ARONSON, 1998). Nesta fase é importante monitorar e controlar o plano proposto pela solução do modelo e, conforme ROSE (1999), isto pode indicar a necessidade de mudanças e redesenho do modelo.

No processo de tomada de decisão é necessário definir o “dono do problema”, delegar responsabilidades aos gerentes responsáveis, ter competência para interpretar os resultados e ter autoridade para implementar o que foi planejado (BIO, 1985).

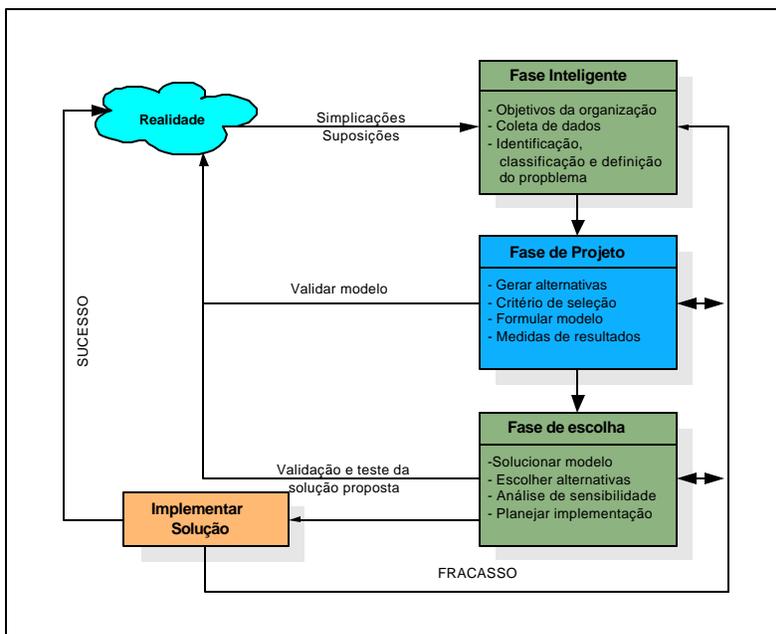


Figura 1 – Estágios do processo de tomada de decisão (TURBAN e ARONSON, 1998).

3. Problemas de manejo florestal

Dada a complexidade da realidade, a tarefa de identificação e definição de um problema de manejo florestal nem sempre é trivial. Muitas vezes, o que é identificado como problema, tal como um custo excessivo, pode ser apenas o

sintoma de um problema, causado, por exemplo, por um nível inadequado de estoque (TURBAN e ARONSON, 1998). Há vários tipos de problemas e muitas vezes uma classificação pode ajudar a identificar a técnica de solução mais adequada. Uma classificação simples, conforme Lentz (1991), citado por SILVA JUNIOR (1993), resulta nas seguintes categorias: problemas bem estruturados, problemas mal estruturados, problemas mal definidos e problemas em contexto múltiplo.

Um problema é bem estruturado quando todas as características que definem as situações insatisfatórias e desejadas são conhecidas, bem como as alternativas de ação disponíveis (STAIR, 1998); em geral são problemas rotineiros das empresas florestais. Por outro lado, um problema é mal estruturado quando as características das situações insatisfatórias e desejadas são conhecidas, mas as alternativas de ações são desconhecidas. Este tipo de problema, em geral, de cunho estratégico, de médio a longo prazo, ocorrem com menor frequência nas empresas (STAIR, 1998). Um problema é mal definido quando há insatisfação com a situação atual, mas o tomador de decisão não consegue definir as características da situação desejada. Muitos problemas de manejo florestal possuem esta característica. Um exemplo é a otimização simultânea de recursos madeireiros e fauna, onde, ao se desejar, produzir madeira e aumentar a população de determinada espécie animal, é necessário conhecer dados de reprodução, crescimento, hábitos alimentares, níveis populacionais adequados e outros, geralmente indisponíveis ou difíceis de obter. Por outro lado, em um problema em contexto múltiplo, o tomador de decisão não é capaz de definir nem mesmo a situação insatisfatória.

O tipo e estrutura dos problemas variam entre e dentro das empresas. Por exemplo, a conversão otimizada de madeira em multiprodutos pode ser um problema bem estruturado em uma empresa, e mal estruturado em outra que porventura desconheça ou não utilize ferramentas adequadas para tal finalidade. As características das estruturas dos problemas e a forma de resolvê-los, sugere que somente com o progresso científico é possível desenvolver novos modelos e tecnologias e melhorar a formação dos tomadores de decisão, tornando os mesmos

capazes de identificar e definir os problemas e as técnicas mais adequadas para resolvê-los.

Os manejadores florestais defrontam-se com diversos tipos de problemas, envolvendo, por exemplo, quantificação de madeira para diferentes finalidades, atendimento de demandas periódicas para diferentes usos da madeira e clientes, quantificação de receitas e custos de empreendimento (análise de fluxos de caixa), planejamento da produção de forma sustentável, atendimento dos múltiplos objetivos conflitantes da floresta, geração e avaliação de alternativas de manejo, posicionamento estratégico na venda ou na aquisição de terras, simulação e avaliação de diferentes regimes de desbastes e outros. Além de identificar e descrever tais problemas, suas soluções envolvem a definição do horizonte de planejamento, e a coleta, processamento e armazenamento de dados e informações, conforme discutido a seguir.

O planejamento da produção florestal requer a avaliação de cursos alternativos de ação que envolvem a definição de um período de validade, cuja duração é o horizonte de planejamento. Segundo REZENDE e OLIVEIRA (1993), o tempo no qual se pode correr riscos em fazer pressuposições para o projeto dependem das circunstâncias particulares de cada situação. Assim, no caso florestal, o período futuro está ligado ao período de obtenção do produto e à exaustão dos investimentos realizados. A escolha de um horizonte curto sacrifica a representatividade da solução, enquanto um horizonte longo pode acarretar um grande número de variáveis (BARROS e WEINTRAUB, 1982). Em horizontes maiores corre-se o risco dos planos futuros serem frustrados devido a mudanças nas condições econômicas, aparecimento de produtos concorrentes, alteração de custos, mudanças tecnológicas e incertezas associadas à obtenção dos dados, dentre outros (REZENDE e OLIVEIRA, 1993). Por outro lado, adotar um horizonte muito curto, resulta em risco de não considerar acontecimentos que podem ocorrer após o período adotado. CLUTTER et al.(1983) sugeriram valores de horizontes de planejamento iguais a 1,5 a 2,0 vezes o comprimento do ciclo de corte da floresta, porém, o tamanho do horizonte depende muito da espécie e do objetivo do manejo.

Em manejo florestal, os cursos alternativos de ação referem-se às alternativas de manejo do problema. Uma alternativa de manejo é uma seqüência de decisões que ocorrem ao longo de um horizonte de planejamento. Tais decisões podem representar, por exemplo, épocas de cortes, plantios, intensidades de desbastes, tecnologias e variedades a serem utilizadas. As diferentes combinações de ocorrência dessas decisões ao longo do horizonte e a interdependência das decisões criam uma infinidade de alternativas de manejo.

Existem duas abordagens clássicas para a identificação e representação das alternativas de manejo para um dado problema, batizadas por Modelo I e Modelo II (JOHNSON e SCHEURMAN, 1977). A estrutura do modelo I foi originalmente proposta por Kidd et. al. (1966); citado por DYKSTRA (1984), recebendo essa designação por JOHNSON e SCHEURMAN (1977). Nessa formulação, a variável de decisão X_{ij} representa a fração da unidade de manejo¹ (*u.m.*) *i* que deverá ser assinalada à prescrição *j*. O fundamento básico dessa formulação é que, inicialmente, a floresta é subdividida em *u.m.'s.* de acordo com um critério preestabelecido de classificação, sendo posteriormente prescrito um elenco pré-determinado de alternativas para cada *u.m.* As prescrições representam seqüências preestabelecidas de ações que ocorrerão ao longo do horizonte de planejamento. As prescrições podem referir, por exemplo, às idades de colheita de uma área, seguida de regeneração. Uma vez que uma dada fração de área seja assinalada a uma dada prescrição de manejo, ela continuará sobre tal prescrição durante todo o horizonte de planejamento, ou seja, a identidade física da *u.m.* se mantém em todo o horizonte de planejamento. O horizonte de planejamento é subdividido em períodos, e as ações ocorrem no início dos mesmos.

A formulação, designada de Modelo II por JOHNSON e SCHEURMAN (1977), permite que *u.m.'s.* distintas colhidas e regeneradas em um mesmo período do horizonte de planejamento sejam reagrupadas em novas *u.m.'s.*, resultando na perda da identidade física das *u.m.'s.* Segundo DAVIS (1987), a abordagem da

¹ Área geograficamente contígua e homogênea em relação às características físicas, vegetativas e de desenvolvimento. Em geral correspondem aos talhões florestais.

regulação florestal² via Modelo II permite uma maior flexibilidade na identificação da seqüência ótima de cortes, resultando em maior flexibilidade na definição de novas *u.m*'s. Diferenças básicas entre estes dois modelos podem ser encontradas em DYKSTRA (1984).

O número de alternativas de manejo geradas para um dado problema depende de fatores como: número de unidades de produção, estrutura dos povoamentos (composição de idades, espécies, regime de manejo), usos da madeira e outras decisões consideradas pelo manejador (aquisição de terras, desbaste, corte raso, variações nas idades de colheita e outras), podendo chegar a centenas ou milhares de alternativas. JOHNSON e SCHEURMAN (1977) relatam problemas com 265.665 alternativas de manejo em países de clima temperado, onde o horizonte de planejamento pode chegar a até 100 anos. No Brasil, apesar dos horizontes serem mais curtos (15 a 30 anos), fatores como o rápido crescimento das florestas, divisão do horizonte em períodos menores (maior detalhamento), capacidade de rebrota dos eucaliptos, diferentes idades de colheita, dentre outros, podem resultar num elevado número de alternativas de manejo.

Os sistemas de suporte à decisão utilizam interfaces específicas para geração de alternativas de manejo com diferentes filosofias. BARROS e WEINTRAUB (1982), desenvolveram uma rotina onde o planejador define explicitamente cada alternativa de manejo, considerando-se apenas de 3 a 5 alternativas para cada *u.m*. Por outro lado, RODRIGUEZ (1988) apresentou um modelo de geração de alternativas de manejo no qual foram consideradas variações nas idades de colheita e no comprimento do ciclo de corte (1, 2 ou 3 rotações). TAUBE NETTO (1984) desenvolveu uma rotina geradora, por meio de matriz de transição, resultando no pacote comercial PlanFlor. Neste pacote, define-se, inicialmente, os estados possíveis para os talhões e, para cada um dos estados, os estados possíveis no ano seguinte (Quadro 1). Através de uma matriz de transição T, cujo elemento T(j,k) é igual a um ou zero, a transição do estado j para o estado k no ano seguinte é liberada ou proibida, respectivamente. Percorrendo-se a matriz convenientemente, o sistema

² Regular uma floresta significa determinar onde, como, quando e quanto produzir de bens e serviços da floresta de modo sustentável para melhor alcançar os objetivos do empreendimento (DAVIS, 1987).

gera para cada estado inicial, todas as seqüências de decisões (as alternativas de manejo).

CARNIERE (1989) apresentou uma metodologia em que as alternativas de manejo são consideradas em uma estrutura de redes, sendo a busca da solução ótima desenvolvida no contexto da programação linear. Com as redes e um algoritmo de solução adaptado foi desenvolvido um programa para problemas de grande porte e a metodologia propiciou uma considerável economia de memória e tempo computacional quando comparado com a rotina do PlanFlor

Outros exemplos de rotinas computacionais geradoras de prescrições de manejo incluem o Rx Write desenvolvido por McDill e Rose (1994), citados por ROSE (1999) e o sistema InFlor-SagFlor desenvolvido pela Universidade Técnica de Lisboa em Portugal (MIRAGAIA et al., 1998; FALCÃO e BORGES, 1999).

Apesar dos avanços nos procedimentos geradores de alternativas de manejo, existem ainda muitas lacunas a serem preenchidas. A maioria das rotinas desconsidera aspectos ambientais no processo decisório e poucas rotinas consideram opções de desbaste. Os objetivos conflitantes da atividade madeireira e as preocupações

Quadro 1 - Exemplos de descrição dos estados possíveis de um projeto em um contexto florestal

Código do Estado	Descrição	Estados Possíveis no ano seguinte
V	Venda de Terra logo após o corte	5
B1	Primeiro corte após plantio seguido pela primeira regeneração (primeira rebrota)	21
B2	Segundo corte após plantio seguido pela segunda regeneração (segunda rebrota)	31
R	Reforma: novo plantio em área cortada	11
I	Implantação: plantio em terreno nativo	11
11	Primeiro ano após I ou R	12
12	Segundo ano após I ou R	13
13	Terceiro ano após I ou R	14
14	Quarto ano após I ou R	15, R, C, V, B1
15	Quinto ano após I ou R	16, R, C, V, B1
16	Sexto ano após I ou R	17, R, C, V, B1
17	Sétimo ano após I ou R	18, R, C, V, B1
18	Oitavo ano após I ou R	R, C, V, B1

21	Primeiro ano após B1	22
22	Segundo ano após B1	23
23	Terceiro ano após B1	24
24	Quarto ano após B1	25, R, C, V, B2
25	Quinto ano após B1	26, R, C, V, B2
26	Sexto ano após B1	27, R, C, V, B2
27	Sétimo ano após B1	28, R, C, V, B2
28	Oitavo ano após B1	R, C, V, B2
31	Primeiro ano após B2	32
32	Segundo ano após B2	33
33	Terceiro ano após B2	34
34	Quarto ano após B2	35, R, C, V
35	Quinto ano após B2	36, R, C, V
36	Sexto ano após B2	37, R, C, V
37	Sétimo ano após B2	38, R, C, V
38	Oitavo ano após B2	R, C, V
0	Aguardando Implantação	0, I, V
5	Projeto vendido (descartado)	5
C	Corte sem plantio no mesmo ano	R, 41
41	Um ano após C aguardando R	42, R
42	Dois anos após C aguardando R	43, R
43	Três anos após C aguardando R	R
6	Projeto aguardando eventual compra	6, I

ambientais requerem critérios de geração ainda pouco conhecidos. Por exemplo, o manejo de uma determinada espécie animal pode requerer grandes áreas (vários talhões), enquanto a produção de madeira muitas vezes requer uma abordagem a nível de um único talhão. Pode haver, portanto, um conflito na definição do que será a unidade de manejo, sendo este um problema que ainda não foi devidamente solucionado.

4. Dados e informações

Os dados são caracteres, imagens ou sons que podem ser pertinentes e utilizáveis para uma tarefa em particular. As informações podem ser consideradas como um conjunto de dados cuja forma e conteúdo são apropriados para uma utilização em particular (CAMPOS FILHO, 1994). A informação, portanto, compreende dados trabalhados ou analisados

(OLIVEIRA, 1993), sendo um dos recursos vitais da empresa, isto é, a base para a tomada de decisão. É exatamente o uso da informação que gera conhecimento na empresa florestal (Figura 2). Pode-se observar que há duas formas de gerar conhecimentos: usando uma informação recebida, ou usando informações advindas da análise de dados. A função principal dos dados é alimentar os diferentes modelos, produzindo novos dados que analisados e interpretados pelos usuários ganham forma e conteúdo transformando-se em informações. Entretanto, uma informação não tem validade se não for usada para gerar conhecimentos. Aqui cabe mencionar uma situação indesejável que ocorre atualmente em muitas empresas: dedicação total aos dados sem a devida transformação dos mesmos em conhecimento. Gerar conhecimento demanda investimento científico, o que nem sempre ocorre quando uma empresa fica presa em sistemas de banco de dados e rotinas operacionais.

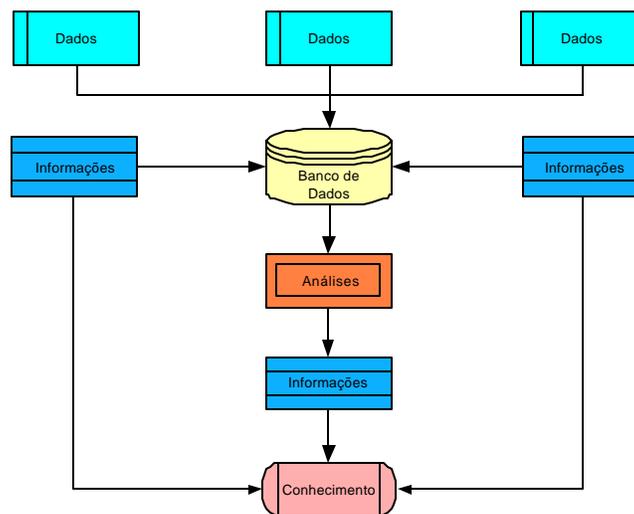


Figura 2 – Geração de conhecimentos através de dados e informações.

De um modo geral, a formulação de problemas utilizando modelos requer a geração, a priori, de cursos alternativos de ação que para avaliá-los, o manejador necessita de dados variados. Os modelos de planejamento requerem, em geral, os seguintes tipos de dados: crescimento e produção da floresta, informações técnicas em manejo florestal, custos e preços de produtos

madeireiros e não madeireiros, políticas e restrições existentes, dentre outros.

Muitos desses dados requeridos não estão disponíveis para o tomador de decisão. Considerando que a aquisição de dados custa tempo e dinheiro, é necessário uma avaliação criteriosa dos seus custos e benefícios para o processo decisório. Segundo ROSE (1999), essa é uma das questões complicadas na coleta de novos dados, pois apesar dos seus custos de aquisição poderem ser bem estimados, os seus benefícios nem sempre são passíveis de medição. Por exemplo, qual o retorno (benefício) proporcionado pela aquisição de dados de levantamento topográfico para alimentação de um sistema de informações geográfica? Neste sentido, apesar de muitos indivíduos considerarem que haverá melhoria no processo decisório, proporcionada pelas informações georeferenciadas, haverá bastante dificuldades de medir este benefício.

Os problemas de manejo de recursos naturais são, por natureza, de longo prazo, envolvendo, portanto, decisões futuras que necessitam de previsão de dados. Os riscos e incertezas do ambiente, aliados às dificuldades de aproximação de uma função eficiente para previsão desses dados, são alguns fatores que dificultam a aquisição de dados de boa qualidade, comprometendo a validade dos resultados dos planos produzidos.

Os dados são transformados em informações através da utilização de métodos científicos (ROSE, 1999), sendo requisitadas em diversos níveis no processo de tomada de decisão (supervisor, coordenador, gerentes, diretores etc). Com o crescimento das empresas, as necessidades de informação tornam-se críticas, especialmente para as pessoas dos níveis mais altos da gerência. Neste sentido, as decisões tornam-se mais complexas e delicadas, os volumes de dados crescem e os administradores afastam-se do ponto onde os acontecimentos ocorrem, dentro da estrutura da organização (BIO, 1985). Além disso, uma informação só é útil se o usuário tiver formação profissional suficiente para interpretá-la e, de fato, utilizá-la.

A geração de informações de qualidade é tida como um benefício básico que pode ser obtido com um planejamento adequado dos sistemas de informação (SI). Por outro lado, o desenvolvimento de SI com base numa abordagem pouco criteriosa, pode transformar a falta de informações em superabundância de

informações irrelevantes. A esse respeito convém transcrever aqui trechos do trabalho de Russel (1967), citado por BIO (1985).

a) *"Forneçam-lhe mais informações": os sistemas de informações são projetados com base na hipótese de que a principal deficiência das condições de atuação dos administradores é a falta de informações relevantes".* De acordo com o mesmo autor, não se pode negar que muitos administradores se ressentem da falta de uma boa quantidade de informações, mas não se deve aceitar que esse seja o principal problema que eles enfrentam em termos de informação. O autor destaca que os administradores padecem mais de uma superabundância de informações irrelevantes.

b) *"O administrador necessita das informações que deseja: muitos dos indivíduos que projetam sistemas de informações 'determinam' as informações necessárias perguntando aos administradores que informações gostariam de ter, pressupondo-se que estes sabem que informações exigem e efetivamente desejam".* Segundo BIO (1985), para um administrador saber de que informações tem necessidade é preciso que esteja ciente de cada tipo de decisão que deve tomar (e realmente toma). Grande parte dos administradores sabem, pelo menos, alguns tipos de decisões que lhes cabem tomar. Entretanto, as suas noções, tendem a ser deficientes em decorrência de um princípio importante de economia científica: *quanto menos se sabe a respeito de um fenômeno, maior é o número de variáveis exigidas para explicá-lo*. Portanto, o administrador não compreende o fenômeno que controla, procura proteger-se e, em relação a informações, ele quer 'tudo'. O indivíduo que projeta sistemas de informações procura fornecer até mais do que tudo, aumentando, assim, o que já era uma carga excessiva de informações irrelevantes".

É difícil caracterizar as necessidades de informações relevantes, se não houver uma compreensão dos processos decisórios (BIO, 1985). O sistema de armazenamento, recuperação e processamento de dados pode determinar o tipo de informações disponíveis em um dado instante. Algumas empresas podem dispor de apenas valores médios de custos das operações, enquanto outras podem dispor dessas informações mais detalhadas, em função do modelo de armazenamento adotado. Algumas vezes, as informações armazenadas não podem ser utilizadas no

processo de tomada de decisão devido às dificuldades do sistema em recuperá-las. Estes aspectos realçam a necessidade de cuidados no armazenamento, processamento e recuperação das informações pelos SI's. Porém, vale ressaltar que o excesso de dados e de informações irrelevantes podem contribuir para decisões equivocadas.

5. Sistemas de suporte à decisão

As pessoas usam informações em seus ambientes de trabalho produzidas por diferentes modelos e processadas por tecnologias variadas. As pessoas são solicitadas, por exemplo, a identificar e formular problemas que ocorrem nas organizações, escolher e construir modelos de suporte à decisão, coletar e introduzir dados nos computadores, interpretar saídas de relatórios e monitorar a soluções propostas, para melhorar a qualidade das decisões tomadas (LAUDON e LAUDON, 1999). As atitudes das pessoas a respeito de seus empregos, empregados ou da tecnologia de computação podem ter um efeito poderoso sobre sua capacidade de usar as informações e as tecnologias de modo produtivo. O papel dos recursos humanos nas organizações são determinantes no uso e sucesso de outros fatores relacionados ao processo decisório. Por exemplo, o emprego de tecnologias para tomada de decisão pode ter pouco sucesso ou ser simplesmente ignorado devido a fatores, como: desconhecimento das técnicas por parte dos gerentes responsáveis pela sua aquisição, benefícios do uso nem sempre reconhecidos pelos níveis gerenciais e falta de pessoal qualificado para execução utilização das tecnologias (CARVALHO, 1997).

Informações, tecnologias e recursos humanos, utilizados no processo decisório, são interdependentes. ROSE *et al.* (1981), apresentou uma revisão sobre esses aspectos e apontou como eles se relacionam e afetam o manejo florestal. O autor identificou situações onde haviam dados adequados, mas com tecnologias inadequadas para processamento dos dados, ou situações onde haviam sofisticados recursos computacionais, modelo e analistas habilidosos, mas uma carência de dados detalhados ou precisos não justificavam o uso daqueles recursos. O autor

observou que freqüentemente justificava-se a inutilidade do planejamento ou pobreza de seus resultados pela falta de dados e, ou tecnologias para manipulá-los. Entretanto, BORGES (1999), adverte que problemas reais existem não pela necessidade de dados e tecnologias, mas com a necessidade de uma especificação adequada do problema para o qual se busca solução. A adequada especificação do problema é obviamente dependente da habilidade das pessoas em identificá-lo e formulá-lo. Os argumentos do autor ressalta, claramente, a importância de se valorizar cada elemento envolvido no processo de tomada de decisão.

A complexidade do processo decisório devido a fatores como necessidade de usos de diferentes modelos para suporte à decisão, grandes volumes de dados a serem coletados, inseridos e processados para produzirem informações de qualidade e na freqüência desejadas para os tomadores de decisão, requer o uso de um sistema que possa auxiliá-lo em todas estas tarefas. Tais sistemas, são comumente designados de sistemas de informação (SI), que são técnicas computacionais utilizadas com objetivos de integrar, recuperar e extrair dados de uma forma ordenada, para fornecer as informações necessárias ao processo de tomada de decisão (TURBAN e ARONSON, 1998). Os SI podem afetar o desempenho das modernas organizações, abrindo-lhes, inclusive, novos espaços e oportunidades de atuação competitiva, motivo pelo qual os gerentes e administradores não podem se furtar a compreender sua natureza e a utilizar seus recursos com eficácia. O leque de novas oportunidades que os SI trazem começa com a melhoria e otimização das operações internas da organização, indo até as suas operações externas, auxiliando na competitividade através de benefícios diretos aos clientes ou usuários (CAMPOS FILHO, 1994). Os SIs geram informações precisas e na freqüência necessária para planejamento e controle (BIO, 1985).

Sistema de suporte executivo, sistemas de informações geográficas, sistema de suporte à decisão individual e em grupo, são alguns exemplos de SI's utilizados para auxiliar o processo decisório. Sistema de inventário florestal e cubagem de árvores, sistemas de controle de cadastro florestal e sistema de suporte à decisão em planejamento florestal, são alguns exemplos de SI's utilizados para dar suporte à tomada de decisão no setor florestal.

Característica, conceitos e detalhes sobre SI de uso universal podem ser encontrados em TURBAN e ARONSON (1998) e LAUDON e LAUDON (1999).

Existem duas categorias de SI's: sistemas naturais e sistemas planejados. Os sistemas naturais resultam da criação de procedimentos isolados, não havendo evidências de planejamento global. Os sistemas planejados, ao contrário, são implantados com um rigoroso planejamento global (BIO, 1985). Segundo este autor, a maioria dos sistemas existentes nas empresas é do tipo *Naturais*, em razão de uma visão equivocada de muitos administradores de que os sistemas naturais são mais baratos. Há também duas categorias de custos em sistemas: o de planejá-los e o de operá-los, sendo que o mais significativo, ao longo do tempo, é o custo de operação. O maior custo de operação em sistemas naturais resulta do maior tempo necessário para organizar as informações requeridas, em função de uma baixa integração entre os seus módulos, da falta de uma definição de quais informações são realmente requeridas, consumindo recursos em processamento de informações desnecessárias, e o dimensionamento inadequado dos recursos tecnológicos em relação ao volume de processamento requerido. Como tais operações são rotineiras nas empresas, ao longo do tempo, o custo de operacionalizar os sistemas naturais tende a ser muito maior do que os custos do sistema planejado (BIO, 1985).

Um SI é uma parte integrante de uma organização e é um produto de três componentes: tecnologia, práticas de trabalho e recursos humanos (Figura 3) com objetivo de produzir informações ao processo decisório (LAUDON e LAUDON, 1999). Segundo estes autores, estas entidades se ajustam mutuamente, e não se pode entender ou usar sistemas de informação em empresas de forma eficiente sem o conhecimento de suas dimensões em termos de organização e de pessoas, assim como de suas dimensões técnicas.

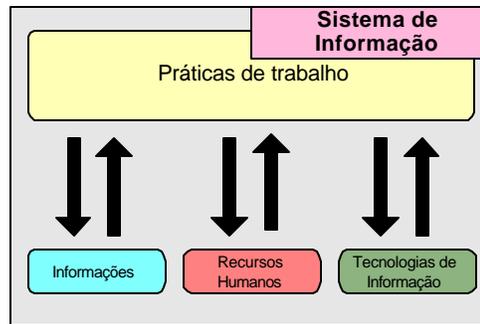


Figura 3 – Sistemas de informação e suas dimensões em termos de informações, recursos humanos, tecnologia e práticas de trabalho.

CAMPOS FILHO (1994) ressalta a necessidade de planejar adequadamente o desenvolvimento de sistemas de informações para as empresas atingirem seus objetivos, evitando frustrações e prejuízos no desenvolvimento de sistema. Segundo LAUDON e LAUDON (1999), um sistema de informação deve ser desenvolvido com o objetivo de integrar informações, recursos humanos, tecnologias de informação e práticas de trabalho. A dimensão informações de um SI envolve o atendimento das reais necessidades de informações dos tomadores de decisão, produzidas na quantidade e frequência desejadas e de boa qualidade. A dimensão pessoas do SI envolve treinamento, atitudes no emprego, ergonomia e interface com o usuário. A dimensão tecnologia dos sistemas de informação consiste em hardware e software de computadores e tecnologias de armazenamento de dados e comunicações adequados para um dado contexto. A dimensão prática de trabalho dos sistemas de informação envolve assuntos como hierarquia das organizações, especialidades funcionais, procedimentos empresariais, cultura e grupos de interesse político. Segundo CAMPOS FILHO (1994), as empresas devem estar atentas para todos os componentes de um sistema de informação, pois se o sistema de informação não for considerado como um produto desses componentes, corre-se o risco delas não conseguirem administrar seus sistemas corretamente. O mesmo autor afirma que o ponto crucial do gerenciamento de um sistema está em como os componentes interagem para criar práticas de trabalho que atendam aos objetivos da organização. Ainda, segundo este autor, uma das causas mais frequentes de

insucesso de um sistema de informação está na tendência de supervalorizar as tecnologias em detrimento as práticas de trabalho.

Dentre os sistemas de informações, os sistemas de suporte à decisão (SSD) têm-se destacado em diversos campos de aplicação, em função das suas potencialidades para o tomador de decisão. Segundo SILVA JUNIOR (1993), os antigos sistemas de informações gerenciais, baseados principalmente em banco de dados, deram lugar aos SSD. Enquanto os sistemas de informações gerenciais permitem apenas aumentar a capacidade de armazenar e acessar dados e realizar cálculos matemáticos, os SSD acrescentam a estes, os mais variados modelos tais como: econômicos, matemáticos e estatísticos, estruturados de maneira a facilitar a análise de alternativas e a interação entre o usuário e o computador. O SSD representa uma evolução no conceito de S.I. uma vez que se propõe a ser flexível e ágil, apoiando-se em métodos operacionais (OLIVEIRA, 1993).

TURBAN e ARONSON (1998), ampliando o conceito de SSD apresentado por Scott-Morton no início da década de 70, conceitua-os como: "um sistema computadorizado interativo e com interface amigável que auxilia o tomador de decisão em todas as fases do processo decisório, utilizando-se de dados, modelos e conhecimento de especialista na solução de problemas mal estruturados". O foco de um SSD está na eficácia da tomada de decisões em face de problemas comerciais não-estruturados e semi-estruturados. Eles oferecem o potencial de gerar maiores lucros, menores custos e melhores serviços, sendo utilizados em todos os níveis das organizações (STAIR, 1998). Um SSD auxilia, mas não substitui julgamentos pessoais, permitindo testar de maneira rápida e objetiva, diferentes estratégias de ação. Este recurso é conhecido na literatura internacional como "*what-if analysis*", ou seja, pode-se analisar a priori os resultados da implantação de certas decisões. Conforme STAIR (1998), esses sistemas permitem manipular grandes volumes de dados, obter e processar dados de fontes diferentes, executar análises e comparações complexas, dão suporte à abordagens de otimização, satisfação e heurística, possuem orientação tanto textual como gráfica e proporcionam flexibilidade de relatórios e de apresentação. Banco de dados, banco de modelos, sistemas de

informações gerenciais ou gerenciador de banco de dados, gerenciador de banco de modelos e interface são os componentes de um SSD (Figura 4).

O cerne de um SSD é composto por um banco de dados, um banco de modelos e um gerador de SSD. Um SSD típico, além disso, contém uma interface com o usuário, uma conexão para banco de dados externos e acesso a outros sistemas baseados em computadores. A interface com o usuário de um SSD, chamada de gerenciador de diálogo, possibilita que os tomadores de decisão acessem e manipulem o SSD com facilidade e permite a utilização de termos e expressões comerciais comuns. O acesso a banco de dados externos propicia que o SSD percorra vastos volumes de informações contidas no banco de dados, permitindo, por exemplo, que o SSD recupere informações sobre estoques, vendas, pessoal, produção, finanças, contabilidade e outras áreas. Por fim, o acesso a outros sistemas baseados em computadores permite que o SSD seja ligado a outros sistemas da empresa.

O desenvolvimento de SSD relacionados ao manejo de recursos florestais, iniciou com o sistema Timber RAM (*Timber Resource Allocation Method*) de NAVON (1971), dando origem a uma série de sistemas usados pelo Serviço Florestal Americano, dentre eles o FORPLAN (*Forest Planning*) (IVERSON e ALSTON, 1986), MUSYC (*Multiple Use-Sustained Yield Calculation Technique*) (JOHNSON et al., 1979), o SPECTRUM (Estados Unidos, 1999, citado por RODRIGUEZ e BORGES, 1999). No Brasil, o primeiro sistema desenvolvido foi o PlanFlor (TAUBE NETTO, 1984) e mais recentemente o SisFlor (RODRIGUES et al., 2000). O SisFlor, por exemplo, oferece uma interface flexível amigável com o usuário, oferecendo opções adicionais ao contemplar desbastes, uso múltiplo e diferentes modelos de silvicultura. O sistema permite trabalhar com diferentes modelos de planejamento, tais como programação linear, programação com múltiplos objetivos e programação inteira. Uma descrição detalhada desse sistema é apresentada no capítulo dois.

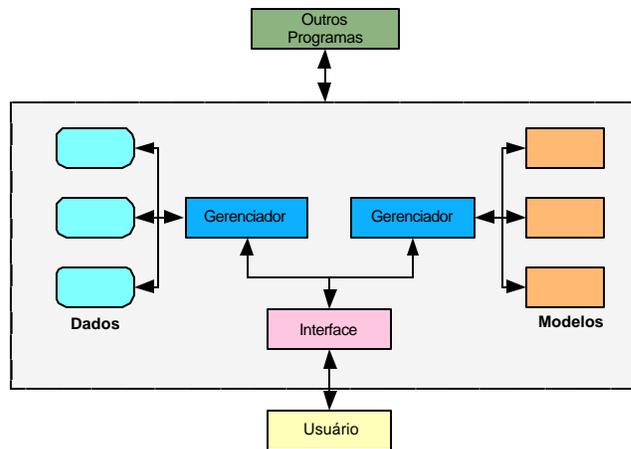


Figura 4 – Componentes básicos de um sistema de suporte à decisão (OLIVEIRA, 1993).

5.1. Modelos utilizados em SSD florestal

Um modelo pode ser definido como uma representação simplificada de fenômenos do mundo real. Na ciência, a utilização de modelos vai desde os modelos em escala reduzida (barragens, topografia, edificações etc), passando por modelos de aviões para estudo de aerodinâmica e estendendo até modelos analíticos de processos físicos e mentais (ANDRADE, 1990). A modelagem de um problema envolve a definição do problema e sua abstração para formas quantitativas e, ou qualitativas. Para um modelo matemático, as variáveis são identificadas e as equações descrevendo suas relações são estabelecidas. Simplificações são feitas, sempre que necessário, através de um conjunto de suposições.

A solução de problemas de manejo florestal envolve a necessidade de usos de muitos modelos. Alguns exemplos são os modelos de prognose da produção florestal, modelos econômicos para avaliação dos planos alternativos, modelos de Pesquisa Operacional e heurísticas para formulação e solução de problemas envolvendo a alocação de recursos, dentre outros. A seguir apresenta-se uma revisão sobre alguns modelos utilizados em SSD florestal.

5.1.1. Modelos de prognose da produção

Estudos sobre modelos de crescimento e de produção florestal iniciaram-se na Europa e foram expandidos na América do Norte, onde atualmente são, largamente, utilizados. CAMPOS e PAULA NETO (1988) definiram tais modelos como um conjunto de relações matemáticas e tabelas, utilizados para descrever quantitativamente os povoamentos florestais.

As técnicas de planejamento necessitam grandemente da capacidade de previsão da produção futura da floresta, daí a necessidade de uso desses modelos.

Existem muitas formas para estimar a produção de povoamentos florestais, variando de simples inventários, a sofisticados modelos de crescimento e produção. O inventário florestal contínuo permite estimar a produção corrente de povoamentos florestais e fazer projeções, para curtos períodos de tempo, sob um único tratamento. Essa prática permite o ajuste de modelos que expressem condições médias do povoamento; entretanto, em alguns casos, pode ser onerosa e não permitir o conhecimento da dinâmica do povoamento. Para estimativas de crescimento e produção a longo prazo, bem como estimativas correntes, visando indicar a melhor opção entre diferentes tratamentos, é mais indicado o uso de modelos ou sistemas preditivos que, além de apontar opções silviculturais, servem para a atualização de inventários florestais. Esses modelos compreendem um conjunto de relações matemáticas e tabelas, que é utilizado para descrever quantitativamente os povoamentos florestais; são de diferentes tipos, sendo a escolha dependente da precisão e da flexibilidade exigida na estimação (CAMPOS e PAULA NETO, 1988).

A necessidade de modelagem de povoamento florestais varia com a intensidade de manejo, sendo que o nível de precisão requerido das estimativas em geral aumenta proporcionalmente com essa intensidade de manejo. Existem modelos de crescimento e produção que utilizam valores do povoamento como unidade básica de modelagem, e modelos que utilizam árvores individuais. São, basicamente, três os tipos de modelos: modelos de povoamento, geralmente inflexíveis para avaliar multiprodutos e grande número de tratamentos, embora possam ser eficiente quando o objetivo se restringe à produção total, modelos de distribuição de diâmetros, que permitem estimar a produção por

classe de diâmetro, e modelos de árvores individuais, de uso mais recente e metodologia em desenvolvimento. A escolha de um desses tipos de modelos deve ser feita com base na exatidão e na flexibilidade proporcionada, no nível de detalhamento exigido e nas práticas de manejo a serem avaliadas (BURKHART, 1979). Exemplos de modelos de crescimento e produção são encontrados em CLUTTER et al. (1983) e em DAVIS e JOHNSON (1987).

Em alguns casos, as práticas de desbastes são imprescindíveis no manejo florestal. Assim, a generalização dos modelos de crescimento e produção para efetuar previsões em povoamentos desbastados é indispensável, dado os novos enfoques e objetivos do manejo florestal moderno, onde práticas de usos múltiplos são cada vez mais comuns. SCOLFORO (1990) e SCOLFORO e MACHADO (1996), utilizaram um conjunto de modelos para estimar o volume e a área basal por unidade de área e, também, a sobrevivência, antes de ocorrer o primeiro desbaste para *Pinus caribaea* var. *hondurensis*. As equações geradas foram utilizadas para gerar informações de crescimento e produção por classe diamétrica, em associação com a função Weibull. Para possibilitar a simulação de desbastes, foram analisadas variáveis como proporção de árvores e de área basal removidas, razões de diâmetro e idade de desbaste, tendo a simulação sido feita pelo “software” simulador de desbastes.

DIAS (2000), avaliou a eficiência do modelo de CLUTTER (1963) em prever o crescimento e a produção em volume e área basal, em povoamentos de eucalipto submetidos à desbaste. O modelo, devidamente avaliado, foi utilizado na construção de tabelas de densidade variável, o que tornou possível fazer simulações de alternativas de desbaste, para diferentes capacidades produtivas. Análises econômicas foram feitas, utilizando dados de custos de desbaste, colheita e condução dos povoamentos, juntamente com dados de preços de madeira. As comparações entre as intensidades de desbaste, definidas individualmente como projetos, foram feitas mediante o emprego do benefício periódico equivalente.

NOGUEIRA (1999) utilizou o método dos ingressos percentuais, desenvolvido por Garcia (1999) citado por NOGUEIRA (1999) para uso em manejo de plantações de eucalipto destinadas a multiprodutos. A época de se

realizar o primeiro desbaste, bem como o intervalo de tempo entre os desbastes sucessivos foi determinada com base na dinâmica de crescimento e no ingresso das árvores em sucessivas classes de diâmetro. A distribuição diamétrica para idades mais avançadas foi feita utilizando-se a função logística e, em seguida, foram calculados os ingressos percentuais das árvores nas classes diamétricas em diferentes idades. De acordo com o autor, a metodologia proposta apresenta algumas vantagens que justificam a sua adoção para determinar a época exata de se realizar o desbaste em plantações de eucalipto, dentre elas: não se baseia em decisões empíricas; não depende da frequência das árvores por hectare; permite determinar a época de iniciar os tratamentos de desbastes pelo menos um ano antes da idade técnica de desbaste; e permite que os seus resultados sejam aplicados corretamente em outros experimentos com características diferentes, como qualidade de sítio e objetivos de manejo. Maiores informações sobre as pesquisas envolvendo desbaste podem ser encontradas em NOGUEIRA (1999) e DIAS (2000).

Em função da necessidade constante de atualização de inventários e o alto volume de informações e processamentos requeridos para prognose da produção, a maioria das empresas utilizam sistemas de informação específico para esta finalidade. Portanto, é fundamental considerar a possibilidade de integração desses sistemas com outros sistemas da empresa, como os Sistemas de Suporte à Decisão.

5.1.2. Modelos econômicos

A elaboração de planos de manejo florestal requer a avaliação econômica de alternativas de manejo. Os critérios empregados para avaliação dessas alternativas são: a produtividade física, o Valor Presente Líquido (VPL), o Benefício (ou Custo) Periódico Equivalente (B(C)PE); o Custo Médio de Produção (CMP), a Taxa Interna de Retorno (TIR) e o Valor Esperado da Terra (VET) (NEWMAN, 1988). A utilização desses critérios deve ser feita observando-se as limitações inerentes a cada um deles. Por exemplo, ao se utilizar o VPL, deve-se atentar para as diferentes durações dos investimentos. Neste caso, o B(C)PE pode corrigir o problema. Por

outro lado, investimentos com duração infinita podem ser avaliados pelo VET. Maiores detalhes sobre este assunto podem ser encontrados em NEWMAN (1988) e RESENDE e OLIVEIRA (1993), dentre outros.

A avaliação econômica das alternativas de manejo requer também considerações sobre o valor do estoque residual da floresta, formado por povoamentos com diferentes estruturas de idades, em função do último corte previsto dentro do horizonte de planejamento. As diferentes idades do povoamento, aliadas a fatores como variações de material genético e produtividade do local, por exemplo, determinam diferentes valores para o estoque residual. Estes diferentes valores, obviamente, devem ser considerados na análise de fluxo de caixa de cada alternativa. Segundo (CLUTTER et al., 1983), a avaliação do valor desse estoque pode ser feita de duas formas distintas:

- acrescentando-se ao cálculo da renda obtida até o último corte de cada alternativa, o valor da renda da madeira em pé ao final do horizonte de planejamento da respectiva alternativa, qualquer que seja a sua idade (preço em potencial daquele lote);
- prolongar o manejo até o final de seu ciclo e acrescentar ao cálculo da renda o valor esperado da terra (VET), o qual representa um ganho de uma seqüência infinita de rotações ótimas, cada qual com seu valor atualizado.

5.1.3. Modelos de regulação florestal

Os primeiros métodos de regulação desenvolvidos, baseavam-se em variáveis do povoamento como a área e o volume. Reckagel, 1917; citado por DAVIS (1987), descreveu 18 métodos de determinação do corte dentro das categorias de controle por área, controle por área modificado e controle por volume. As diferenças básicas entre esses métodos estão relacionadas com a principal variável de controle: área florestal ou volume de madeira (DAVIS, 1966 e MEYER *et al.*, 1961). Esses métodos derivam do conceito teórico de

florestas normais³ e são questionáveis quanto a sua aplicabilidade (THOMPSON, 1966). Segundo WARE e CLUTTER (1971), a inadequação desses métodos ocorre porque: na maioria das técnicas, assume-se a existência de uma floresta padrão com uma estrutura ideal, e a transformação de uma floresta que hoje apresenta uma situação específica, em uma floresta padrão é o objetivo do planejamento; várias técnicas estão baseadas no conceito de estoques de distribuição normal de pouca pertinência em condições reais; poucos métodos fazem uso de dados detalhados de volume, taxas de crescimento e potencial de produção; as técnicas antigas de manejo ignoram considerações econômicas; e, nenhum dos procedimentos é flexível na seleção das unidades de manejo a serem exploradas.

Segundo CLUTTER *et al.* (1983), em lugar da concepção anterior deve-se sustentar que o real papel do manejador é gerenciar de modo inteligente, estruturas florestais não estáveis ou não balanceadas. Problemas de colheita sempre levam à necessidade de maximizar o retorno financeiro ou minimizar o custo e atendimento de certas restrições que influenciam nas decisões, levando à necessidade de uso de técnicas de análises mais sofisticadas, como os modelos de Pesquisa Operacional (PO).

Surgida durante a segunda guerra mundial, a Pesquisa Operacional (PO) tomou grande impulso na década de 50 com o surgimento dos computadores de alta velocidade. Aplicada inicialmente na área militar, foi difundida para o setor industrial e governamental e a seguir para a agropecuária. Atualmente pode-se encontrar aplicações da PO em quase todas as áreas da atividade humana ((HILLIER e LIEBERMAN, 1988). Segundo PAULA JUNIOR (1998), PO é o nome dado a um conjunto de modelos e algoritmos destinados a determinar o melhor curso das ações que visam garantir o funcionamento ótimo de sistemas, sob restrições de recursos escassos. As aplicações da PO envolvem pelos menos dois aspectos: construir

³ Florestal normal pode ser entendida como aquela que possui uma série de povoamentos equiprodutivos com idades variando desde a mais jovem até a mais velha, que coincide com a idade de rotação (LEUSCHNER, 1984).

descrições ou modelos matemáticos, econômicos e estatísticos de problemas de decisão para tratar situações de complexidade e incerteza, e analisar as relações que determinam as conseqüências futuras prováveis de alternativas adotando medidas apropriadas de eficácia de modo a calcular o mérito relativo de cada uma dessas ações (WAGNER, 1986). Existem diversas técnicas de PO, entre elas a programação linear (PL), programação inteira (PI), programação com múltiplos objetivos (PMO) são as mais utilizadas no tratamento de problemas de manejo florestal, em especial, da regulação da produção.

A PL é útil para encontrar a(s) melhor(es) solução (ões) de problemas que tenham seus modelos representados por expressões lineares. A sua grande aplicabilidade e simplicidade deve-se à linearidade dos modelos (BREGALDA, 1988). Muitos classificam o desenvolvimento da PL entre os mais importantes avanços científicos dos meados do século vinte. O *Simplex*, o primeiro algoritmo para solução de problemas de PL foi idealizado por Koopman para solução de problemas em redes e a seguir adaptado por Dantzig em 1947 para problemas de PL, sendo atualmente a técnica de solução mais usada.

O uso da PL para resolver problemas de planejamento florestal, em especial de regulação da produção, teve início na década de 60 (CURTIS, 1962; LOCKS, 1964 e NAUTYAL e PEARSE, 1967). Entretanto, os primeiros trabalhos que consagraram o uso extensivo dessa técnica no setor florestal foram os de NAVON (1971), trabalhando no serviço florestal americano e WARE e CLUTTER (1971) na Universidade da Georgia, em colaboração com a indústria de celulose do sul dos EUA. Exemplos de usos da PL em regulação da produção, no Brasil, são encontrados em TAUBE NETTO (1984), RODRIGUEZ e MOREIRA (1989), RODRIGUEZ *et al.* (1986), RODRIGUES (1997) e RODRIGUES *et al.* (1998), RODRIGUES *et al.* (1999) e SILVA (2001).

O número de bens e serviços comerciais e não-comerciais, oferecidos pelas florestas vêm crescendo ano após anos, e são cada vez mais importantes na tomada de decisão em propriedades públicas e privadas. A integração de valores ambientais dentro do uso múltiplo, em planos de manejo florestal, tem ganhado atenção especial nos últimos anos. O interesse da sociedade e os valores relacionados com os

produtos e serviços oferecidos pelas florestas foram recentemente traduzidos dentro de novas ações, leis, e regulamentações (TARP *et al.*, 1997). Esses novos requerimentos ampliaram os objetivos dos planos de manejo florestal. As novas exigências do setor demandaram também novas abordagens para tratamentos dos problemas.

Modelos de PL, até então tidos como adequados para solução dos problemas do setor, passaram a ser questionados em função de suas limitações no tratamento de problemas envolvendo múltiplos objetivos. Assim, modelos mais adequados para tratamentos desses problemas como a Programação com Múltiplos Objetivos (PMO), ganharam a atenção dos manejadores.

Desenvolvida para planejamento de uso múltiplo, a PMO foi primeiramente introduzida na literatura florestal por FIELD (1973). Desde então, vários autores têm aplicado esta técnica em uma variedade de problemas florestais. A PMO é uma técnica útil na solução de problemas de planejamento em que vários objetivos conflitantes devem ser otimizados de modo simultâneo. Práticas de uso múltiplo podem resultar neste tipo de problema e alguns exemplos de uso são encontrados em FIELD (1973), BARE e MENDOZA (1988); BARE e MENDOZA (1992), MENDOZA (1985), MENDOZA *et al.* (1986), RUSTAGI e BARE (1987), HOTVEDT *et al.* (1982) e TARP *et al.* (1997).

O primeiro sistema a incorporar a análise do uso múltiplo foi o Timber RAM (*Resource Allocation Method*), desenvolvido pelo Serviço Florestal dos Estados Unidos (NAVON, 1971). Segundo RIBEIRO (1995), apesar desse sistema ter sido desenvolvido para auxiliar no planejamento de recursos públicos, ou no planejamento de terras para uso múltiplo, ele possui forte tendência para uma alocação prioritária de madeira. Um outro sistema com uma estrutura de modelagem mais adequada para o planejamento florestal envolvendo múltiplos objetivos é o MUSYC (*Multiple Use-Sustained Yield Calculation Technique*) (JOHNSON *et al.*, 1979). Além de contemplar aspectos madeireiros, este sistema incorpora objetivos não-madeireiros, contribuindo para melhor prática do uso múltiplo da floresta.

Apesar dos avanços na pesquisa, na prática, o uso da PMO tem sido bastante reduzido, em especial no Brasil, devido ao desconhecimento da técnica e de seus

possíveis benefícios, por parte de alguns manejadores. Outra razão é a carência de interfaces computacionais para a formulação e solução de problemas envolvendo a utilização destes modelos. Os problemas reais, além de envolver muitos objetivos e restrições, só podem ser formulados e resolvidos com usos de sistemas automatizados. Nenhum dos sistemas de suporte à decisão desenvolvidos, até então, no Brasil utilizam modelos de PMO como suporte à tomada de decisão.

A utilização de PMO eliminou uma das limitações dos modelos de PL ao permitir o tratamento e otimização simultânea de objetivos conflitantes. Entretanto, assim como os modelos de PL, a PMO não é adequada para tratamento de problemas envolvendo variáveis inteiras, uma vez que um dos pressupostos desses modelos é a continuidade nas variáveis de decisão. As mudanças de paradigmas no planejamento de recursos naturais alteram os critérios de sustentabilidade dos planos de manejo florestal, com conseqüente impacto na forma de modelar e resolver os novos desafios. Por exemplo, o planejamento estratégico, utilizava como indicadores de sustentabilidade, os fluxos de madeira ou níveis da existência de volume em cada classe de idade, passíveis de solução por modelos de PL. Este tipo de modelagem apesar de permitir a análise econômica de várias soluções, apresenta algumas limitações por não permitir considerações de interações espaciais das intervenções sobre os povoamentos.

Restrições de corte evitando abertura de extensas clareiras e, ou fragmentação excessiva da paisagem são alguns critérios de sustentabilidade ecológica traduzidos nos novos paradigmas em manejo recursos florestais (BORGES, 1999). Uma outra restrição imposta aos planos de manejo florestal é a alocação de uma única alternativa de manejo em cada unidade de produção, sendo atualmente conhecida como restrição de singularidade. Esta restrição é extremamente útil para contornar o problema de inoperância das soluções propostas por formulações via PL e PMO. A propriedade de continuidade nos valores das variáveis de decisão nesses modelos permite que mais de uma alternativa de manejo seja alocada para uma mesma unidade de manejo. Ao considerar as unidades de manejo como unidades mínimas de produção, não permitindo subdividi-las em novas unidades, os planos resultantes tornam-se inoperantes.

O problema da restrição de singularidade designando a alocação de uma única alternativa de manejo à cada talhão tem sido bastante estudado. Uma das alternativas para resolver tais problemas consiste em utilizar os modelos lineares discretos. A principal alternativa de formulação é o uso de modelos de PI, que incorporam variáveis binárias, $X_{ij} \in (0,1)$, em sua estrutura, conforme encontrado em JOHNSON e TEDDER. (1983), YOSHIMOTO e BRODIE (1993), MURRAY (1999) e TORRES-ROJO e BRODIE (1990).

A restrição envolvendo considerações espaciais no manejo de ecossistemas florestais é um outro problema também bastante estudado. Tais restrições são definidas em termos de limitações de adjacência, onde o corte simultâneo de unidades vizinhas é proibido. Diversos trabalhos têm sido feitos utilizando-se formulações via modelos de PI com variáveis binárias (0, 1) para resolver esses problemas (TORRES-ROJO e BRODIE, 1990; YOSHIMOTO e BRODIE, 1993; MURRAY; 1999).

Um dos resultados da incorporação de restrições de singularidade e adjacência nos modelos é que os problemas de planejamento da colheita florestal, freqüentemente formulados como problemas de programação linear, tornam-se de natureza combinatória. Enquanto os problemas lineares contínuos podem ser resolvidos com o algoritmo *Simplex*, de forma eficiente, os problemas lineares discretos, salvo alguns casos particulares, carecem da mesma sorte. O estudo de Garey e Johnson (1979), citados por GOLDBARG e LUNA (2000) foi um marco para o entendimento dos obstáculos que se interpõem entre uma possível solução teórica para um problema de programação discreta e sua implementação prática através dos instrumentos computacionais da atualidade. Segundo GOLDBARG e LUNA (2000), a dificuldade da abordagem exata dos problemas denominados NP-árduos está na explosão combinatória dos métodos enumerativos. De acordo com IGNIZIO e CAVALLER (1994), os algoritmos para solução de problemas de PI mais utilizados, "*branch and bound*", "*implicit enumeration*" e "*cutting plane*", são aplicáveis apenas em problemas de tamanho moderado. Esse certamente não é o caso das aplicações florestais que freqüentemente exigem a definição de milhares de variáveis de decisão.

Em virtude dessa realidade, nos últimos anos tem-se verificado o surgimento de um significativo conjunto de técnicas e algoritmos computacionais muito eficientes, embora não garantam a solução ótima para os problema de PI. Esses algoritmos são denominados de *heurísticos* ou *aproximativos* (GOLDBARG e LUNA, 2000). No capítulo 3 é apresentado um estudo mais detalhado de algumas heurísticas mais difundidas com aplicações em diversos setores, inclusive, com algumas aplicações no setor florestal.

5.2. Alguns equívocos no uso de modelos

O fato dos modelos serem representações simplificadas do mundo real sugere, que nenhum modelo, permite prever corretamente a realidade para o qual se propõe representar. Entretanto, isto não pode ser utilizado como a justificativa para não se usar o modelo (ROSE, 1999). De acordo com este autor, *“não se deve subestimar o conhecimento acumulado de florestais experientes no campo de tomada de decisões em manejo, mas também não devemos superestimar a habilidade do cérebro humano para entender e ordenar complexas interações e suas conseqüências em tomadas de decisão”*. Segundo o mesmo autor, os riscos dessa abordagem está na limitada habilidade humana para entender complexas interações em sistemas, conduzindo para decisões potencialmente pobres. Além disso, no planejamento florestal, tem-se que conviver com as conseqüências de decisões pobres por um longo prazo. As propriedades dos modelos lhes conferem vantagens quando utilizados em processos de tomada de decisão, por exemplo, ajuda a organizar o conhecimento servindo como banco de dados, propiciando maior facilidade para atualizar e manipular informações. A modelagem é mais rápida, menos dispendiosa e de menor risco do que a experimentação em sistemas reais. Os modelos permitem medir os impactos das decisões sobre o sistema e testar as estratégias de decisão sem causar danos à empresa, à organização, aos clientes ou ao meio ambiente, além de permitir prever conseqüências futuras (TURBAN e ARONSON, 1998 e ROSE, 1999). Entretanto, segundo TURBAN e ARONSON

(1998), os modelos apresentam algumas desvantagens no auxílio à tomada de decisão, tais como:

- as diversas opções de modelos, podem levar os tomadores de decisão a gastarem muito tempo decidindo que modelo deve ser usado;
- em alguns casos, os modelos não prevêm sistemas reais com exatidão, conduzindo a resultados falsos ou a erros;
- alguns modelos exigem um alto grau de sofisticação matemática, tornando-os muito difíceis de construir e os resultados difíceis de interpretar.
- muitos administradores não confiam nos modelos que utilizam complexos algoritmos e manipulações matemáticas

Muitos equívocos têm surgido na utilização dos modelos, alimentando argumentos infundados para alguns manejadores justificarem a sua não utilização no suporte à tomada de decisão. Um primeiro equívoco diz respeito a algumas visões distorcidas de alguns indivíduos de que os modelos em teoria são bons, mas na prática falham. Esta afirmação é absurda pois a construção de um modelo decorre da existência de um problema para o qual se pretende encontrar solução. A adequação do modelo decorre precisamente do sucesso da aplicação prática e não existem modelos bons em teoria e maus na prática, eles são validados com base na comparação entre solução que oferece e a realidade que se pretende representar. Neste caso, o modelo deve ser confrontado com o problema florestal real e, se for de fato inadequado, deve ser redesenhado (BORGES, 1999). A este respeito, ROSE (1999), salienta que quanto mais o manejador estiver ciente das limitações dos modelos mais fácil será evitar rejeição e abusos dos mesmos.

Um segundo equívoco é o argumento de que os modelos exigem um grande volume de dados, e portanto, não são úteis em muitas situações porque há escassez ou indisponibilidade de dados (BORGES, 1999). De acordo com Tavares *et al.*, (1996), citado por BORGES (1999) é precisamente em situações onde os dados e a informações são escassos que urge a necessidade de desenvolver modelos e sistemas que dêem apoio à decisão. Este equívoco resulta, com freqüência da incapacidade em formular modelos adequados a situações em que os dados são escassos, (ROSE,1999).

Um outro argumento equivocado é que os modelos são úteis em outros campos, mas não se conformam com alguns problemas florestais específicos. Segundo BORGES (1999), este equívoco diz respeito à suposta especificidade das técnicas de desenho de modelos. Esta afirmação confunde a informação técnica, que é de fato específica, com o desenho de um modelo. As técnicas de construção de modelos como programação linear, programação inteira, programação dinâmica e outras, são de utilização universal. Este equívoco resulta com frequência da pouca experiência de alguns manejadores na construção de modelos adequados à especificidade dos problemas.

Outra filosofia equivocada é a ameaça dos modelos em substituir a função de alguns manejadores. A este respeito, BORGES (1999), salienta que os modelos não substituem a experiência e o bom senso. Pelo contrário, eles têm o objetivo principal de oferecer informação e de reforçar a experiência. O modelo não decide, ele ajuda a decidir e isto só é possível se houver competência para interpretar a sua solução.

5.3. Tecnologia e o processo decisório

Entende-se por tecnologias de informação, o conjunto de hardware e software que desempenha uma ou mais tarefas de processamento das informações, como coletar, transmitir, estocar, recuperar, manipular e exibir dados. Neste grupo, estão incluídos microcomputadores (em rede ou não), mainframes, scanners de código de barra, estações de trabalho, *software* de execução, *software* de planilhas eletrônicas ou de banco de dados, coletores de dados etc (CAMPOS FILHO, 1994). A tecnologia é um fator indispensável ao processo decisório. Por exemplo, o bom desempenho de um modelo requer a disponibilidade de dados com níveis de detalhes adequados, de boa precisão e livres de tendências, mas nenhum volume elevado de informações e, ou a complexidade de muitos processamentos envolvidos na construção de um modelo para auxiliar o processo de tomada de decisão, podem ser factíveis com uma quantidade de tempo razoável, se uma tecnologia adequada não for utilizada.

As tecnologias para suporte à decisão, em maior ou menor grau de sofisticação, estão presentes em todas as fases do processo decisório. A grande diversidade de tecnologias disponíveis, as diferentes tecnologias exigidas em função da natureza e complexidade dos problemas de decisão, o custo/benefício das tecnologias para uma dada aplicação, e preferência ou cultura do tomador de decisão ou da organização no uso de tecnologias, são alguns fatores que influenciam a escolha do pacote tecnológico para auxiliar no processo de tomada de decisão.

Excluindo a preocupação de listar as inúmeras alternativas tecnológicas para auxílio à tomada de decisão, na Figura 5, são listadas algumas tecnologias de suporte à decisão e suas interações com as fases do processo decisório. As Redes Neurais Artificiais (RNA), por exemplo, podem auxiliar na análise de sistemas complexos, produzindo informações mais confiáveis, como previsão de preços e demanda de determinados produtos. *Data Mining* e *Online Processing Analysis* (OLAP) são úteis no processamento e manipulação de dados utilizados como *inputs* do problema, oriundos de fontes externas e internas do sistema.

Problemas podem ser detectados e auxiliados por especialistas de alto escalão através dos Sistemas de Suporte Executivo (SSE). A fase de projeto e escolha pode incluir a participação de grupos de tomadores de decisão, que por sua vez apresentam melhor sinergismo quando auxiliados por Sistema de Suporte à Decisão em Grupo (SSDG). Pacotes comerciais personalizados, como geradores de modelos, otimizadores e heurísticos são indispensáveis na fase de escolha do processo decisório. A fase de implementação pode ser auxiliada através da participação de grupos de tomadores de decisão, auxiliados novamente pelos SSDG. Os Sistemas de Informações Geográficas (SIG's) oferecem enormes potenciais para análises e apresentação de soluções alternativas. Este sistema por permitir o georeferenciamento da informação, auxilia, por exemplo, na análise espacial e escolha das soluções propostas. Por exemplo, em planos alternativos de manejo florestal, um plano que apresente uma maior fragmentação da paisagem ou que permita aberturas de extensas clareiras poderiam ser descartados. Sistemas de Suporte à Decisão (SSD) e Sistemas Especialistas (SE) são outros exemplos de tecnologias alternativas para auxiliar no processo decisório, e conforme o esquema

proposto na Figura 5 estas duas tecnologias podem auxiliar em todas as fases do processo.

Scanners de código de barra, estações de trabalho, *software* de planilhas eletrônicas, computadores ligados em redes ou não, e inúmeros outros recursos, os quais não foram ilustrados no esquema da Figura 5, podem ser utilizados como auxílio ao processo decisório.

A tecnologia (em especial os computadores e os *softwares*) têm papel de destaque na análise de sensibilidade em estágios anteriores e precedente ao planejamento; teste de diferentes cenários, comuns em planejamento florestal, execução de processamento em tempo até então impossível de se efetuar; constituem-se na forma mais barata e eficiente de armazenar e recuperar informações (ROSE, 1999).

As tecnologias constituem-se num elemento indispensável dentro do processo decisório, mas o seu uso isolado não garante a geração de bons planos. Em alguns casos, é necessário avaliar, por exemplo, se o mesmo recurso gasto em treinamento de pessoas não produziria resultados superiores a simples aquisição de tecnologias (ROSE, 1999). Por si só, o computador não assegura que a empresa passe a contar com sistemas de alta qualidade. Ao mesmo tempo, porém, sem o seu emprego, certos benefícios objetivados com o planejamento de sistemas podem não ser factíveis (BIO, 1985).

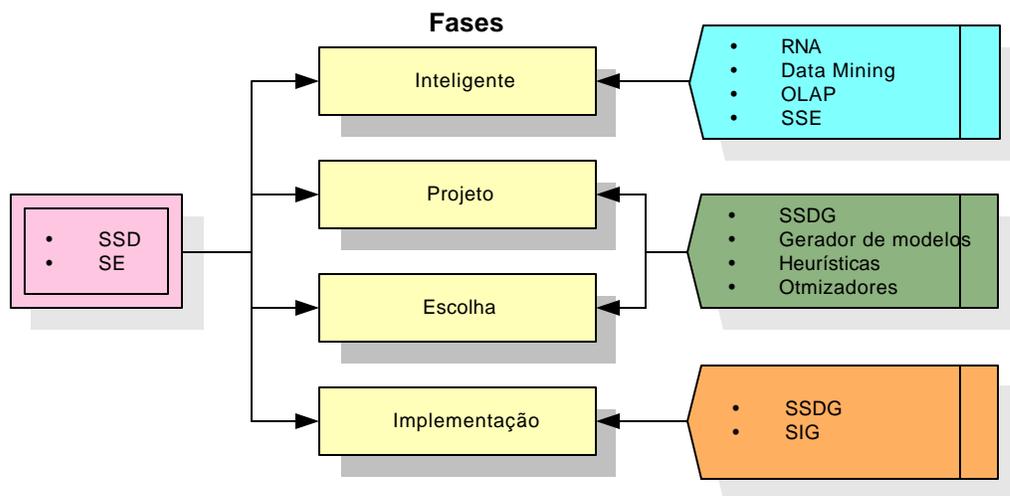


Figura 5 - Exemplos de tecnologias alternativas no processo decisório: adaptado de TURBAN e ARONSON (1998).

Muitas organizações florestais reconhecem a contribuição essencial que a tecnologia moderna pode fazer para melhorar tomadas de decisões em planejamento. Existem, contudo, algumas barreiras que necessitam ser vencidas para facilitar as inovações tecnológicas em processo de planejamento florestal (ROSE, 1999), sendo:

- em muitos indivíduos existe uma resistência natural a mudanças e inovações;
- os gastos com recursos financeiros e tempo para treinamento de pessoal são às vezes elevados;
- elevado tempo de treinamento de pessoal para lidarem com a nova tecnologia, desestimula a substituição de tecnologias já consagradas;
- necessidade de reciclagem e especialização dos manejadores para lidarem com a nova tecnologia.
- perda de autoridade dos manejadores em função da necessidade de trabalho em equipe, já que muitas tecnologias são de caráter multidisciplinar.

6. CONSIDERAÇÕES FINAIS

Com base na revisão bibliográfica e na discussão apresentada, as seguintes conclusões são pertinentes:

- A utilização de abordagens científicas através do uso de modelos e tecnologias, produzindo informações com qualidade e rapidez, aliados aos papéis dos recursos humanos, são indispensáveis no planejamento florestal;
- Apesar da grande sofisticação de alguns modelos de Pesquisa Operacional utilizados em planejamento florestal, algumas de suas limitações impedem seus usos em grande escala. Um desses problemas, envolve a necessidade de impor restrições de inteireza em variáveis de decisão. Estas restrições incorporam maiores complexidades aos problemas, dificultando suas soluções através dos algoritmos exatos convencionais, dentre eles, o algoritmo *branch and bound*;
- Apesar do avanço da pesquisa e aplicações de diversos modelos de suporte à decisão, seus usos têm sido ainda incipientes, principalmente no Brasil. O desconhecimento das técnicas constitui-se num dos entraves;
- Os sistemas de informação sofreram enormes mudanças nos últimos anos, culminando com sistemas de informações automatizados denominados sistemas de suporte à decisão;
- Apesar das potencialidades inquestionáveis dos sistemas desenvolvidos até então, existem ainda muitas limitações que necessitam ser vencidas. A incapacidade de muitos sistemas em formular planos de manejo considerando conceitos e práticas modernas do manejo florestal, incluindo questões ambientais, silviculturais, econômicas, por exemplo, limitam a utilização desses sistemas em muitos problemas reais. Outra limitação desses sistemas refere-se ao tratamento de problemas envolvendo formulações via Programação Inteira. A solução desses modelos esbarra em limitações dos algoritmos tradicionais de solução (*branch and bound*, *implicit enumeration* e *cutting plane*) para problemas de grande porte.
- Embora seja importante o uso das tecnologias de informação para suporte à decisão no planejamento florestal, não se deve subestimar o conhecimento

acumulado dos manejadores experientes, nem tão pouco subestimar o papel dessas tecnologias, induzido por falhas e equívocos do próprio manejador. Muitas barreiras enfrentadas no tratamento de problemas de planejamento florestal não são devidas à falta de tecnologias ou de dados para resolvê-los, e sim, pela carência de pessoal qualificado para formular e resolver problemas utilizando tais recursos.

- A carência, a qualidade e níveis de detalhes adequados dos dados podem ser apontados também como a causa de muitas falhas quanto a utilização das modernas tecnologias em planejamento florestal. Portanto, não se deve diminuir a importância da base de informação que resulta da investigação multidisciplinar.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, E.L. Introdução à pesquisa operacional - Métodos e modelos para a análise de decisão. **Rio de Janeiro. ed. Livros Técnicos e Científicos, 1990. 377 p.**

BARE, B.B.; MENDOZA, G.A. A soft optimization approach to forest land management planning. **Forest Science**, Washington, 18 (1), 545-552. 1988.

BARROS, O.E., WEINTRAUB, A. Planning for vertically integrated forest industry. **Operations Research**, Baltimore, v.30, n.6, p.1168-1182, 1982.

BIO, S.R. **Sistema de informação: um enfoque gerencial.** São Paulo. Atlas, 1985.183p.

BORGES, J.G. Paradigmas, tecnologias e equívocos em gestão de recursos florestais... **Revista Florestal**, Lisboa, 12(1), p.27-34, 1999.

BREGALDA, P.F. Introdução à programação linear. **Rio de Janeiro, Campus, 1988. 329p.**

BURKHART, H.E. **Growth and yields of southern pine state of the Art.** Dep. of For. Virginia Polytechnic inst. and State Univ., 11p., 1979.

CAMPOS FILHO, M.P. Os sistemas de informação e as modernas tendências da tecnologia e dos negócios. **Revista de administração de empresas**, São Paulo, 34 (6), p.33-45. 1994

CAMPOS, J.C.C.C.; PAULA NETO, F. Experiências em inventário e mensuração florestal. In: **I Simpósio de Pesquisa na UFV.** Viçosa, Universidade Federal de Viçosa, p. 80-81. 1988.

- CARNIERI, C. **Planejamento florestal otimizado via redes de manejo**. Campinas, SP: UNICAMP, 1989. 144p. Tese (Doutorado em Engenharia Elétrica) - Universidade Estadual de Campinas, 1989.
- CARVALHO, R.A.; COSTA, H.G. Contribuição das novas tecnologias de informática à manufatura integrada por computador. In: ENCONTRO NACIONAL DE ENGENHARIA DA PRODUÇÃO, 17, 1997, Gramado. **Anais**. Porto Alegre: Universidade Federal do Rio Grande do Sul, 1997, p.340-346.
- CLUTTER, J.C., PIENAAR, L.V., BRISTER, G.H. *et al.* **Timber management: a quantitative approach**. 3, ed. New York: John Wiley, 1983. 333p.
- CLUTTER, J.L. Compatible growth and yield models for loblolly pine. **Forest Science**, Washington, 9(3), p.354-371. 1963.
- CURTIS, F.H. Linear programming in the management of a forestry property. **Journal of Forestry**, v. 60, p611-616, 1962
- DAVIS, K.P. **Forest management: regulation and valuation**. 2, ed. New York: McGraw-Hill, 1966. 519p.
- DAVIS, L.S., JOHNSON, K.N. **Forest management**. 3. ed. New York: McGraw-Hill, 1987. 780p.
- DIAS, A.N. **Modelagem e avaliação econômica de plantações de eucalipto submetidas a desbastes**. Viçosa, MG: UFV, 2000. 109p. Dissertação (Mestrado em Manejo Florestal) - Universidade Federal de Viçosa, 2000.
- DYKSTRA, D.P. **Mathematical programming for natural resource management**. New York: McGraw-Hill, 1984. 318p.
- FALCÃO, A.; BORGES, J.G. **Sagflor – Simulador de Alternativas de Gestão. Manual de utilização**. Lisboa, 37p, 1999.
- FIELD, D.B. Goal programming in forest management. **Forest Science**, Washington, 19, 125-135. 1973.
- GOLDBARG, M.C.; LUNA, H.P.L.** Otimização combinatória e programação linear: modelos e algoritmos **Rio de Janeiro, Campus, 2000. 649p.**
- HILLIER, F.S., LIEBERMAN, G.J. **Introdução à pesquisa operacional**. Rio de Janeiro: Campus, 1988. 805p.
- HOTVEDT, J.E.; LEUSCHNER, W.A.; BUHYOFF, G.J. A heuristic weight determination procedure for goal programs used for harvest scheduling models. **Can J. For. Res.** 12: 292-298, 1982.
- IGNIZIO, J.P.; CAVALIER, T.M. **Linear Programming**. Englewood Cliffs: Prentice Hall, 1994. 666p.
- IVERSON, D.C. ALSTON, R.M. **The genesis of FORPLAN a historical and analytical review of Forest Service planning models**. USDA Forest Service General Technical Report, INT-214: p1-25, 1986.

- JOHNSON, K. N., SCHEURMAN, H. L. Techniques for prescribing optimal timber harvest and investment under different objectives - discussion and synthesis. **Forest Science**, Washington, 18(1), p.1-31, 1977.
- JOHNSON, K. N., TEDDER, P.L. Linear programming vs. binary search in periodic harvest level calculation. **Forest Science**, Washington, 29(3), p.569-581, 1983.
- JOHNSON, K.N.; JONES, D.; DANIEL, B. **A user's guide to multiple use sustained yield scheduling calculation (MUSYC)**. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Timber Management; 1979. 242 p.
- LAUDON, K.C.; LAUDON, J.P. **Sistemas de informação**. Rio de Janeiro, Livros Técnicos e Científicos. 1999. 389p.
- LOUCKS, D.P. The development of an optimal program for sustained-yield management. **Journal of Forestry**, v.62, p.485-490, 1964.
- MEGGINSON, C.L., MOSLEY, C.D., PIETRI, H.P. **Administração: Conceitos e aplicações**. Harbra: 1986. 544p.
- MENDOZA, G.A. A heuristic programming approach in estimating efficient target levels in goal programming. **Can J. For. Res.** 16: 363-366, 1985.
- MENDOZA, G.A.; BARE, B.B.; CAMPBELL, G.E. Multiobjective programming for generating alternative: a multiple-use planning example. **Forest Science**, Washington, 33(2): 458-468. 1986.
- MEYER, H.A., RECKNAGEL, A.B., STEVENSON, R.A. *et al.* **Forest management**. New York: Ronald, 1961. 282p.
- MIRAGAIA, C.; BORGES, J.G.; TOMÉ, M. Infor, um sistema de informação em recursos florestais. **Revista Florestal**, Lisboa, 12 (1), p51-56, 1999
- MURRAY, A.T. Spatial restrictions in harvest scheduling. **Forest Science**, Washington, 45(1): 45-52. 1999.
- NAUTIYAL, J.C.; PEARSE, P.H. Optimizing the conversion to sustained-yield - a programming solution. **Forest Science**, Washington, 13, p.131-139, 1967.
- NAVON, D.I. **Timber RAM ... a long-range planing method for comercial timber lands under multiple-use management**. Berkeley, CA: U.S. Department of Agriculture, Forest Service, Pacific Southwest Forest and Range Experiment Station, 1971. 22p. (USDA. Forest Service, Res. paper PNW - 70)
- NEWMAN, D.H. The optimal forest rotation: a discussion and annotated bibliography. **General Technical Report**, Washington, 90(48), p.1-47. 1988.
- NOGUEIRA, G.S. **Determinação da idade técnica de desbaste em plantações de eucalipto**. Viçosa, MG: UFV, 1999. 62p. Dissertação (Mestrado em Ciência Florestal) - Universidade Federal de Viçosa, 1999.
- OLIVEIRA, D.P.R. **Sistemas de informações gerenciais**. 2.ed. São Paulo, Atlas: 1993. 267p.

- PAULA JUNIOR, G.G. **Introdução à pesquisa operacional**. Campus, RJ: UENF/CCT, Laboratório de Ciências de Engenharia/Setor de Engenharia da Produção. 1998, 355p. (Notas de aula).
- RESENDE, J.L.P., OLIVEIRA, A.D. **Avaliação de projetos florestais**. Viçosa, MG: Imprensa Universitária, UFV, 1993. 49p. (Universidade Federal de Viçosa, 327)
- RIBEIRO, C.A.A.S. **The economics and regulation of short rotation coppice forest systems**. Colorado: Colorado State University, 1995. 132p.
- RODRIGUES, F.L. **Regulação de florestas equiâneas utilizando programação linear**. Viçosa, MG: UFV, 1997. 109p. Dissertação (Mestrado em Manejo Florestal) - Universidade Federal de Viçosa, 1997.
- RODRIGUES, F.L.; LEITE, H.G.; ALVES, J.M. *SisFlor* – um sistema de suporte à decisão em planejamento florestal. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p974-995.
- RODRIGUES, F.L.; LEITE, H.G.; SILVA, M.L.; GOMES, A.N. Determinação de estratégias ótimas de reforma, condução da brotação e compra de terras, utilizando programação linear. **Revista Árvore**, Viçosa, 23(2), p.169-186. 1999.
- RODRIGUES, F.L.; LEITE, H.G.; SOUZA, A.L.; RIBEIRO, C.A.A.S.; SILVA, M.L. Regulação de florestas equiâneas utilizando programação linear: uma aplicação da teoria do modelo II. **Revista Árvore**, Viçosa, 22(2), p.193-213, 1998.
- RODRIGUEZ, L.C.E. GERADOR – **Forest Management Matrix Generator**. Piracicaba, ESALQ/USP, 1988.
- RODRIGUEZ, L.C.E., LIMA, A.B.N.P., BUENO, A.D. *et al.* Programação Linear no planejamento florestal: uma aplicação prática. **Silvicultura**, São Paulo, 11(41), p.163-168, 1986.
- RODRIGUEZ, L.C.E., MOREIRA, R.M. Gerenciamento de florestas de *Eucalyptus* com modelos de programação linear. IPEF, Piracicaba, 6(19), p.1-15, 1989. (Série técnica).
- RODRIGUEZ, L.C.E.; BORGES, J.G. Técnicas matemática para determinação de níveis sustentáveis de produção florestal: um exemplo em eucalipto. **Revista Florestal**, Lisboa, 12 (1), p83-92, 1999.
- ROSE, D.W. Data, models and technology in forest planning. **Revista Florestal**, Lisboa, 12(1), p35-44, 1999
- ROSE, D.W.; GREGERSON, A.R.E.; HOGANSON, H.M. Planning-with minimum data and technology. 1981
- RUSTAGI, K.P.; BARE, B.B. Resolving multiple goal conflicts with interactive goal programming. **Can J. For. Res.** 17: 1401-1407, 1987.

- SCOLFORO, J.R.S. Sistema integrado para predição e análise presente e futura do crescimento e produção, com otimização de remuneração de capitais, para *Pinus caribea var. hondurensis*. Curitiba, PR: UFPR, 1990. 289p. tese (doutorado em Manejo Florestal) - Universidade Federal do Paraná, 1990.
- SCOLFORO, J.R.S., MACHADO, S.A. Um sistema de crescimento e produção com simulador de desbaste. **Ciência Florestal**, n.50, p.51-64, 1996.
- SILVA JUNIOR, A.G. **Sistemas de suporte à decisão integrado a sistemas especialistas: uma aplicação para gerenciamento de fazendas produtoras de leite**. Viçosa, UFV, 1993. 94p. (Tese mestrado)
- SILVA, G.F. **Problemas no uso de programação matemática e simulação em regulação florestal**. Viçosa, MG: UFV, 2001. 89p. Tese (Doutorado em Manejo Florestal) - Universidade Federal de Viçosa, 2001.
- STAIR, R. **Princípios de sistemas de informação**. Rio de Janeiro. Livros Técnicos e Científicos 1998. 432p.
- STOONER, J.A.F., FREEMAN, R.E. **Administração**. Prentice-Hall: 1985. 538p.
- TARP, P.; PAREDES, G.L.; HELLES, F. A dual approach to policy analysis in multiple-use forest management planning. **Can J. For. Res.** 27: p849-858, 1997.
- TAUBE NETTO, M. Um modelo de programação linear para planejamento de florestas de eucalipto. **Pesquisa Operacional**, Campinas, 4(1), p.19-39, 1984.
- TAVARES, M.C. **Planejamento estratégico. A opção entre sucesso e fracasso empresarial**. Ed. Harbra, São Paulo, 1991. 199p.
- THOMPSON, E.F. Traditional forest regulation model: an economic critique. **Journal Forestry**, Washington, 64(11), p.750-752, 1966 .
- TORRES-ROJO, J.M., BRODIE, J.D. Adjacency constraints in harvest scheduling: na aggregation heuristic. **Can J. For. Res.** 20: 978-986, 1990.
- TURBAN, E; ARONSON, J.E. **Decision support systems and intelligent systems**. New Jersey: Prentice-Hall, 1998. 890p.
- WAGNER, H.M. **Principles of Operations Research**. Englewood Cliffs, N.J: Prentice-Hall, 1986. 851p.
- WARE, G.O.; CLUTTER, J.L. A mathematical programming system for the management of industrial forests. **Forest Science**, Washington, 17(3), p.428-445, 1971.
- YOSHIMOTO, A.; BRODIE, D. Comparative analysis of algorithms to generate adjacency constraints. **Can. J. For. Res.** 24: 1277-1288, 1993.

CAPÍTULO 2

SISFLOR: UM SISTEMA DE SUPORTE À DECISÃO PARA O PLANEJAMENTO DE FLORESTAS EQUIÂNEAS

RESUMO - A complexidade dos problemas envolvidos no planejamento florestal, o excessivo número de variáveis, a necessidade de respostas às diversas questões em um curto espaço de tempo, a necessidade do uso de diferentes modelos e o elevado volume de dados para alimentar tais modelos são alguns fatores que justificam a necessidade do uso de Sistemas de Suporte à Decisão (SSD). Com facilidade para manipular e processar grandes volumes de dados, executando complexas análises através de modelos de otimização e simulação, os SSDs contribuem para a utilização de abordagens científicas para a tomada de decisão. Apesar das inúmeras vantagens da utilização dessa tecnologia no manejo florestal, seu uso tem sido ainda pouco difundido em muitos países, em parte, devido à escassez de programas computacionais adequados. Este trabalho teve como objetivo desenvolver e apresentar uma primeira versão de um SSD para o gerenciamento florestal. O sistema desenvolvido, designado como *SisFlor* (Sistema de Suporte à Decisão Otimizada em Planejamento Florestal) mostrou ser uma alternativa tecnológica viável para auxiliar o tomador de decisão no gerenciamento florestal. Além de sua interface amigável, o sistema permite uma enorme flexibilidade na formulação e solução, através de abordagens científicas, de diversas questões envolvidas no gerenciamento florestal.

Palavras-chave: Gerenciamento Florestal, Sistema de Suporte à Decisão, Pesquisa Operacional.

SISFLOR: A DECISION SUPPORT SYSTEM FOR EVEN-AGE FOREST PLANNING

ABSTRACT - The complexity of problems involved in forest planning, the excessive number of variables, the need for answers to several issues in a short time, the need for the use of different models and the large amount of data to feed such models are some factors that justify the use of Decision Support Systems (DSS). With easiness to manipulate and to process large amounts of data, executing complex analyses via optimization models and simulation, DSS contribute to the use of scientific approaches for decision making. In spite of the countless advantages of applying this technology in forest management, its use has still been little spread in many countries, partly due to the lack of appropriate computational programs. The objective of this work was to develop and present a first version of a DSS for forest management. The system developed, designated as SisFlor (Support System to Optimized Decision in Forest Planning), was shown as a viable technological alternative to assist the decision maker in forest management. Besides its friendly interface, the system allows a great flexibility in the formulation and solution of several issues involved in forest management through scientific approaches.

Key-words: Forest Management, Decision Support System and Operational Research.

1. INTRODUÇÃO

A necessidade de utilização de diferentes modelos, aliada à complexidade e porte dos problemas de planejamento florestal demandam ferramentas sofisticadas para auxiliar em tarefas como processamento de dados, construção de modelos e apresentação de resultados. Tais operações sugerem, obviamente, que sistemas automatizados possam auxiliar com êxito os tomadores de decisão. Os Sistemas de Suporte à Decisão (SSD) baseados em computadores, são indispensáveis nessa tarefa. Em geral os SSD são sistemas computadorizados, interativos e amigáveis que auxilia o tomador de decisão em todas as fases do processo decisório, utilizando-se

de dados, modelos e conhecimento de especialista na solução de problemas mal-estruturados (TURBAN, 1998). Por apresentar como filosofia: a integração de dados, modelos, tecnologias e recursos humanos, os SSD contribuem para uma melhor estruturação dos problemas, melhorando o processo decisório. As características gerais dos SSDs tais como: manipulação de grandes volumes de dados; obtenção e processamento de dados de fontes diferentes; flexibilidade de relatórios; execução de análises e comparações complexas e sofisticadas, através de pacotes de software avançados; suporte à abordagens de otimização, satisfação e heurística e execução e análises de simulação (STAIR, 1998), coloca-os como ferramentas indispensáveis no auxílio à solução de muitos problemas de planejamento florestal.

O desenvolvimento de SSD relacionados a manejo de recursos florestais, iniciou com o sistema Timber RAM (*Timber Resource Allocation Method*) de NAVON (1971), dando origem a uma série de sistemas usados pelo Serviço Florestal Americano, dentre eles o FORPLAN (*Forest Planning*) (IVERSON e ALSTON, 1986), MUSYC (*Multiple Use-Sustained Yield Calculation Technique*) (JOHNSON et al., 1979), o SPECTRUM (Estados Unidos, 1999, citado por RODRIGUEZ e BORGES, 1999) e o PLANFLOR (TAUBE NETTO, 1984). Entretanto, a utilização desses sistemas em larga escala nas empresas florestais brasileiras têm sido ainda muito incipiente. Fatores como desconhecimento dessas ferramentas ou mesmo falta de reconhecimento dos seus potenciais por parte dos tomadores de decisão contribuem para pouca difusão dessas ferramentas nas empresas florestais. Outros fatores como o custo do sistema, adequação à realidade florestal brasileira, interface com o usuário, flexibilidade e rapidez na formulação e solução dos problemas, têm contribuído para a pouca difusão desses sistemas nessas empresas.

Este trabalho teve como objetivo principal desenvolver uma primeira versão de um sistema de suporte à decisão alternativo para auxiliar o tomador de decisão no manejo de florestas equiâneas com enfoque na realidade florestal brasileira. Os objetivos específicos foram:

- apresentar os princípios básicos do sistema de suporte à decisão desenvolvido, destacando sua flexibilidade e facilidade de uso para formulação e solução de diversos problemas de planejamento florestal de grande porte,
- formular e resolver um problema utilizando o sistema desenvolvido.

2. MATERIAIS E MÉTODOS

2.1. Desenvolvimento do sistema

O SisFlor foi desenvolvido utilizando a ambiente de programação *delphi* 5.0. A primeira versão do sistema apresentada neste trabalho utiliza a filosofia de banco de dados *desktop* através do banco de dados *paradox*. O sistema pode ser rodado em *Windows* NT, 95/98. Para uma boa performance do sistema recomendam-se os recursos computacionais mínimos: Microputador com processador 300 Mb, memória RAM de 64 Mb e disco rígido de 10 Gb.

Os principais módulos e funções do sistema foram apresentados através de fluxograma e desenhos de algumas de suas telas de apresentação, com uma rápida explanação desses esquemas, conforme apresentados no item 3 deste capítulo.

2.2. Estudo de caso

Estudo de caso descrito no problema a seguir teve como principal objetivo ilustrar as principais limitações dos modelos de Programação Linear (PL) e Programação Inteira (PI), e assim justificar os desenvolvimentos dos capítulos seguintes. Excluiu-se desse estudo de caso, o propósito de explorar todas as potencialidades do sistema SisFlor em formular e resolver problemas envolvendo, por exemplo, desbastes dos povoamentos, usos múltiplos da madeira, substituições de material genético, aquisição e vendas de terras etc.

2.2.1. Definição do problema

Um povoamento de *Eucalyptus grandis* distribuído em uma área total de 809,00 ha com capacidade produtiva média das unidades de manejo igual a 30,

deverá ser manejado nos próximos dez anos para produção de madeira para celulose. A floresta será manejada com um ciclo de até dois cortes, o que significa que até uma segunda rotação será permitida, ou seja, regimes de alto-fuste (AF) e primeira talhadia (T1) poderão ser prescritos. As rotações poderão variar entre 5 e 7 anos para ambos os regimes de corte. A madeira colhida e vendida no campo (eliminado custos de transportes) será avaliada a um preço único de 20,00 \$/m³. O objetivo do empreendedor é maximizar o valor presente líquido do empreendimento para os próximos dez anos, a uma taxa de juros de 7% ao ano. A demanda de madeira anual para os próximos dez anos não deverá exceder os 45000 m³, porém nunca será menor do 30000 m³. Além disso, o proprietário prevê que poderá reformar, anualmente, no máximo 150 ha. O proprietário deseja obter informações anuais para os próximos dez anos de forma a planejar estrategicamente sua produção. Outras informações do problema são descritas a seguir.

2.2.2. Unidades de manejo

Os critérios de classificação territorial do proprietário estabeleceu a divisão da área total em talhões com diferentes áreas, idades e regimes de manejo, conforme Quadro 2.

Quadro 2 – Classificação das unidades de manejo

Talhão	Área (ha)	Idade (anos)	Regime
1	50,00	1	AF
2	20,00	2	AF
3	25,00	3	T1
4	54,00	4	T1
5	30,00	5	T1
6	45,00	6	AF
7	42,00	5	T1
8	32,00	4	AF
9	26,00	3	T1
10	41,00	2	T1
11	45,00	1	T1
12	49,00	2	AF
13	52,00	3	AF
14	57,00	4	T1
15	60,00	5	AF

16	55,00	6	AF
17	35,00	5	T1
18	52,00	4	AF
19	39,00	3	T1
TOTAL	809,00		

em que: AF = alto-fuste e T1 = primeira talhadia

2.2.3. Estimativa da produção

A equação utilizada para estimativa da produção de *Eucalyptus grandis* foi a mesma utilizada por RODRIGUES (1997), sendo:

$$\ln(\hat{V}) = 6,81842724 - 163,0489370 / I \bar{S} \quad (2)$$

em que:

V = volume total com casca (m³/ha);

I = Idade (anos);

S = capacidade produtiva;

\ln = logaritmo neperiano.

Assumindo uma capacidade produtiva média para todos os talhões de 30 e uma redução de 20% da produção no regime de talhadia, obtêm-se a tabela de produção para o povoamento de *Eucalyptus grandis*, conforme Quadro 3.

Quadro 3– Tabela de produção para *Eucalyptus grandis* em diferente idades e regime de corte

Idade (anos)	Capacidade produtiva (S)	Produção (m ³ /ha)	
		Alto-fuste	Talhadia 1
5	30	308,41	246,73
6	30	369,66	295,73
7	30	420,73	336,59

2.2.4. Custos das operações florestais

As operações florestais consideradas no estudo de caso e seus respectivos custos encontra-se no Quadro 4, extraído do trabalho de SILVA (1991). Estes custos representam os custos médios das operações florestais. O custo da terra foi

considerado como aluguel ou juros sobre o investimento, sendo, portanto, um custo anual. A expressão desse custo pode ser encontrada em SILVA (1991).

Quadro 4 - Resumo dos custos das principais operações, considerando um adicional de 15% atribuído ao custo de administração

Operações	Custo - \$/ha	Ano de ocorrência
Implantação	1040,12	0
Manutenção	97,84	0
	168,85	1
	79,65	2
	79,65	3
	40,25	Do 4 ^o até o ano do primeiro corte
	140,13	No ano do 1 ^o corte
	140,13	No ano do 2 ^o corte
Condução da brotação	40,25	Demais anos
Reforma	871,27	Ano zero do 2 ^o corte
Colheita (\$/m ³)	10,00 [#]	Nos anos do 1 ^o e 2 ^o cortes
Custo de aquisição da terra(\$/ha)	1200,00*	Nos anos de plantio
Custo anual da terra	1200 x i	Anualmente

Já incluído o custo de administração;

* Valor hipotético.

2.2.5. Formulação do problema

O problema foi formulado utilizando-se dois modelos de Pesquisa Operacional (Programação Linear e Programação Inteira), e conforme já mencionado, tem objetivos de comparar e ressaltar algumas limitações desses modelos.

2.2.5.1. Programação linear

Utilizando-se a formulação via modelo I (JOHNSON e SCHEURMAN, 1977), a variável de decisão X_{ij} representa a fração da i-ésima unidade de manejo que deverá ser assinalada à j-ésima prescrição. Após subdivisão da floresta em

unidades de manejo, prescreve-se um conjunto de alternativas de manejo (X_{ij}) para cada unidade. As prescrições representam seqüências preestabelecidas de ações que ocorrerão ao longo do horizonte de planejamento. A estrutura matemática desse modelo para o problema proposto é descrita a seguir:

- **Função objetivo:**

$$Max Z = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} \quad (3)$$

$$C_{ij} = \sum_{k=0}^{H-1} \frac{(R_k - C_k)}{(1+r)^k} \quad (4)$$

C_{ij}^1 = é o valor presente líquido (VPL) de cada hectare da alternativa de manejo j se implementada na unidade de manejo i ;

X_{ij} = variável de decisão representado a área (ha) da j -ésima alternativa de manejo quando assinalada na i -ésima unidade de manejo;

M = número de unidades de manejo ou estratos;

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

R_k = receita total no ano k $\{k = 0, 1, 2, \dots, H-1\}$;

C_k = custo total no ano k , $\{k = 0, 1, 2, \dots, H-1\}$;

r = taxa de juros anual;

H = tamanho do horizonte de planejamento (anos).

Sujeito a:

- **Restrição de área**

Esta restrição determina que a soma das áreas alocadas para cada uma das alternativas de manejo não exceda a área máxima disponível em cada unidade de manejo.

¹ Não considera as diferentes durações dos investimentos, com devidas correções, conforme discutidas em RODRIGUES (1997).

$$\sum_{i=1}^M \sum_{j=1}^N X_{ij} \leq A_i \quad (5)$$

em que:

A_i = área da i -ésima unidade de manejo ($i=1,2,\dots,M$)

- **Demanda de madeira**

Esta restrição estabelece níveis desejáveis de produção em cada período. Há diversas formas de defini-la, em função do objetivo do manejador. Por exemplo, para limitar a produção de um único produto entre um limite inferior e superior, conforme estabelecido no problema, define-se as seguintes expressões:

$$\sum_{i=1}^N \sum_{j=1}^M V_{ijk} X_{ij} \geq D \min_k \quad (6)$$

$$\sum_{i=1}^N \sum_{j=1}^M V_{ijk} X_{ij} \leq D \max_k \quad (7)$$

em que:

V_{ijk} = volume por hectare (m^3/ha) na i -ésima unidade de manejo manejada segundo a j -ésima alternativa de manejo no período k ($k=0, 1, 2, \dots, H-1$);

$D \min_k$ = demanda mínima produzida no período k ($k=0, 1, 2, \dots, H-1$);

$D \max_k$ = demanda máxima produzida no período k ($k=0, 1, 2, \dots, H-1$);

Para o problema proposto os valores de $D \min_k$ e $D \max_k$ foram considerados como sendo 30.000,00 e 45.000,00 m^3 , respectivamente.

A produção ou a demanda periódica pode ser definida para outras finalidades, como: estabelecimento de níveis de flutuações mínimas e máximas periodicamente, produção não declinante, produção para multiprodutos etc.

- **Áreas reformadas periodicamente**

Pode ser de interesse do manejador limitar as áreas de plantio ou reforma, periodicamente. O manejador pode atuar, ainda, no controle dessas áreas impondo-

se cotas das áreas a serem reformadas em cada período. Por exemplo, para impor uma área máxima de reforma periodicamente, pode-se utilizar a seguinte restrição.

$$\sum_{i=1}^M \sum_{j=1}^N a_K X_{ij} \leq AR_k \{K = 0, 1, \dots, H - 1\} \quad (8)$$

em que:

AR_k = área máxima a ser reformada no período k ,

$a_k = 1$, se X_{ij} apresenta reforma no período k ; 0 caso contrário.

O valor de AR_k para o problema proposto foi de 150 ha. Outras restrições como orçamento, mão-de-obra, regulação, caixa, operacional etc, podem ser incorporadas ao modelo de programação, bastando que o manejador as defina como restrições do problema em questão.

2.2.5.2. Programação inteira

A utilização dos modelos de Programação Inteira (PI) é comum em diversos problemas de planejamento florestal. Restrições de singularidade, adjacência e habitat de animais silvestres, são algumas aplicações que requerem a utilização desses modelos. A limitação da PL para estas aplicações deve-se à propriedade de continuidade das variáveis de decisão destes modelos. Ao contrário, na PI as variáveis de decisão são discretas, o que confere a estes modelos um leque de aplicações muito importante em diversos problemas de planejamento florestal.

A formulação de modelos de PI é bastante simples e pode ser obtida a partir dos modelos de PL com algumas alterações, conforme mostradas a seguir:

- **Função objetivo**

A função objetivo do modelo de PI pode ser obtida a partir da função objetivo do modelo de PL, alterando-se apenas o cálculo do coeficiente da função objetivo, assim:

$$\text{Max ou Min } Z = \sum_{i=1}^M \sum_{j=1}^N D_{ij} X_{ij} \quad (9)$$

sendo:

$$D_{ij} = A_i C_{ij}, \quad (10)$$

em que:

D_{ij} = é o valor presente líquido total (\$) de cada alternativa de manejo i , aplicada à unidade de manejo j .

Sujeito a:

- **Restrição de área ou de singularidade**

A restrição de área do modelo de PI pode ser formulada para permitir que apenas uma alternativa de manejo seja alocada em cada unidade de manejo. Esta restrição tem sido comumente chamada de restrição de singularidade e pode ser formulada da seguinte forma:

$$\sum_{i=1}^M \sum_{j=1}^N X_{ij} \leq 1 \quad (11)$$

$$X_{ij} \in \{0,1\}$$

em que:

$$X_{ij} = \begin{cases} 1, & \text{se a prescrição } j \text{ for adotada no talhão } i \\ 0, & \text{caso contrário} \end{cases}$$

- **Restrição de produção**

A diferença desta restrição em relação à sua forma no modelo de PL é o termo A_i (área da i -ésima unidade de manejo) que deve ser incorporado no lado esquerdo da expressão de produção, alterando os coeficientes tecnológicos da restrição de V_{ijk} (m³/ha) para $A_i V_{ijk}$ (m³). Para o problema proposto, tem-se:

$$\sum_{i=1}^N \sum_{j=1}^M A_i V_{ijk} X_{ij} \geq D \min_k \quad (12)$$

$$\sum_{i=1}^N \sum_{j=1}^M A_i V_{ijk} X_{ij} \leq D \max_k \quad (13)$$

- **Restrição de reforma**

A expressão para definição da restrição de reforma no modelo de PI é derivada da sua respectiva expressão no modelo de PL por inclusão do coeficiente tecnológico A_i , assim:

$$\sum_{i=1}^M \sum_{j=1}^N A_i a_k X_{ij} \leq AR_k \{K = 0, 1, \dots, H - 1\} \quad (14)$$

Os níveis das restrições de demandas mínima e máxima e reforma máxima utilizadas nesta formulação foram os mesmos empregados na formulação via PL, o que propicia uma comparação adequada entre as duas formulações.

3. RESULTADOS E DISCUSSÃO

3.1. O Sistema de suporte à decisão

O sistema computacional denominado por SisFlor (Sistema de Suporte à Decisão Otimizada para Planejamento de Florestas Equiâneas) foi desenvolvido em quatro módulos: módulo de informações, módulo de modelagem, módulo gerador de prescrições e módulo de relatórios (Figura 1)

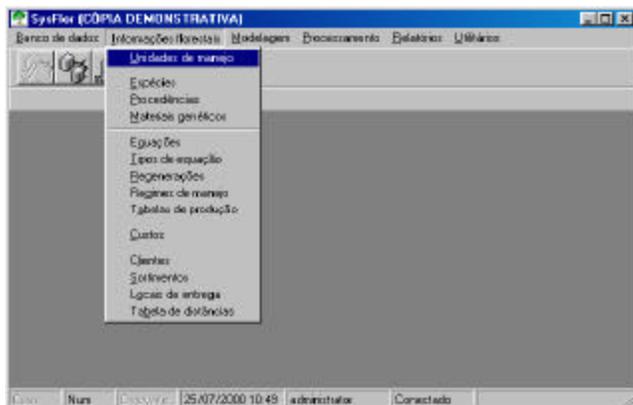


Figura 1 – A interface do sistema de suporte à decisão SisFlor.

Cada um dos módulos principais foi subdividido em outros módulos secundários, conforme apresentados no fluxograma da Figura 2.

As funções dos módulos principais e secundários do SisFlor são descritas, resumidamente, a seguir.

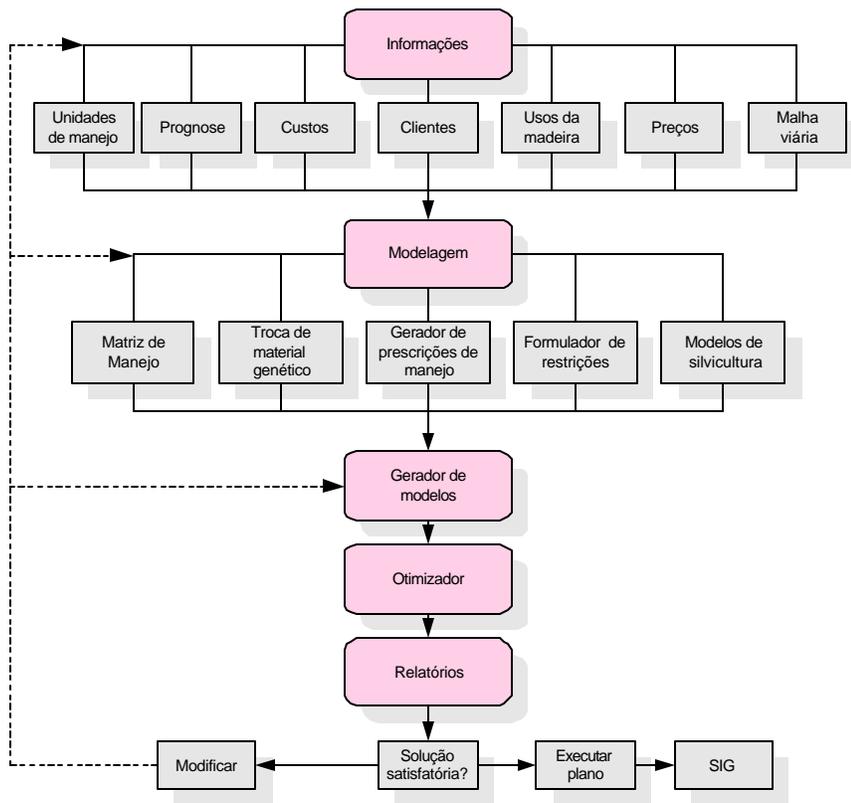


Figura 2– SisFlor: detalhamento dos módulos do sistema.

3.1.1. Módulo para entrada de dados e informações

Neste módulo são armazenados todas os dados e informações requeridos para solução do problema. A estrutura e função dos principais módulos que compõem o Módulo Informações são descritos a seguir.

3.1.1.1. Unidades de manejo

As unidades de manejo (UM) são caracterizadas através de um conjunto de variáveis que descrevem a situação de cada uma das UM no início do horizonte de planejamento (Figura 3). Por exemplo, a UM 7 codificada como A, possui uma área

de 100,5 ha, material genético *Eucalyptus grandis* (codificado como GRAN), manejada atualmente para produção de celulose (CELULOSE), regime inicial de alto-fuste (AF) e idade inicial de 1 ano. A madeira de cada UM pode ser vendida "em pé" ou "colhida" (estas alternativas são representadas pela variável "Tipo-Venda"). Cada UM pode ter sofrido a intervenção de "corte raso" ou desbaste" (a variável "Ult. Corte" representa estas duas possíveis intervenções na UM). Para designar o tempo transcorrido do último desbaste na UM, utilizou-se a variável "Ult. desbaste", representando o tempo transcorrido (anos) após o último desbaste ter sido efetuado na UM. Finalmente, uma variável "estado inicial" deve ser definida neste módulo. Esta variável é utilizada na geração das alternativas de manejo. Maiores detalhes sobre esta variável são apresentados no item 3.1.2.1 deste capítulo, descrevendo a matriz de manejo do sistema.

Unitade	Código	Zonável	M.G. variável	Manejo	Regime	Idade	Estado inicial	Venda	Regeneração	Ult. Desbaste	UB.Code
7 a	100,5	GRAN	CELULOSE	AF	1	Colhida	REFORMA	1	Carateco		
8 b	200	GRAN	CELULOSE	T2	2	Colhida	IMPLANT	1	Carateco		
9 c	50	GRAN	CELULOSE	T1	2	Colhida	REFORMA	2	Carateco		
10 d	70	GRAN	CELULOSE	T1	2	Colhida	CONDICAO	2	Carateco		
11 e	90	GRAN	CELULOSE	T2	1	Colhida	IMPLANT	3	Carateco		
12 f	100	GRAN	CELULOSE	T2	1	Colhida	IMPLANT	3	Carateco		
13 g	85	HIB	CELULOSE	T1	2	Colhida	REFORMA	4	Carateco		
14 h	45	HIB	CELULOSE	T1	2	Colhida	REFORMA	5	Desbaste		
15 i	95	GRAN	CELULOSE	T2	2	Colhida	CONDICAO	6	Desbaste		
16 j	110	URO	CELULOSE	T1	1	Colhida	IMPLANT	7	Desbaste		

Figura 3 - SisFlor: módulo para registro das unidades de manejo.

3.1.1.2. Regime de manejo

A capacidade de brotação da maioria dos eucaliptos confere a este um ciclo de vida especial. Árvores originárias de plantios ou reforma via sementes ou estacas podem ser colhidas na primeira rotação e através de brotações de suas cepas crescerem e produzirem por uma ou mais rotações. Portanto, na evolução de um povoamento de eucalipto durante seu ciclo de corte, pressupõe variações nos seus regimes de manejo (ou de corte). Ao povoamento que se encontra em primeira rotação atribui-se o regime de alto-fuste, onde indivíduos são regenerados via plantio ou reforma, através de mudas provenientes de sementes ou estacas. Ao povoamento que esteja em segunda rotação, onde os indivíduos são regenerados

pela primeira brotação das cepas, atribui-se o regime de primeira talhadia, e assim sucessivamente. Além desses regimes de manejo clássicos, outros regimes especiais para representar, por exemplo, uma área aguardando implantação ou compra, podem ser definidos. O sistema possui grande flexibilidade na definição dos regimes de manejo, podendo ser definido um número grande deles. Na Figura 4 são apresentados alguns desses regimes.



Regime	Sigla	Comentário
0	AFuste	Alto fuste
1	T1	Talhadia 1
2	T2	Talhadia 2
3	T3	Talhadia 3
4	AI	Aguardando implantação
5	AC	Aguardando compra

Figura 4 - SisFlor : módulo regimes de manejo.

3.1.1.3. Material genético

Neste módulo são cadastrados todos materiais genéticos (espécies, clones, procedências etc) incluídos no plano de manejo. Os materiais genéticos desta lista incluem aqueles manejados no momento atual, bem como aqueles que eventualmente no futuro possam vir a ser manejados em substituição a materiais genéticos indesejáveis. Por exemplo, clones de baixa produtividade ou com características tecnológicas indesejáveis, devido a mudanças de objetivo da empresa podem vir a ser substituído por novos clones mais produtivos, de melhor qualidade e que atenda às novas exigências, como diferentes propriedades tecnológicas. O módulo de material genético do SisFlor possibilita a inclusão de diversos materiais genéticos no sistema, bastando cadastrá-los, apresentando seu código, descrição e um comentário ou observação (Figura 5).

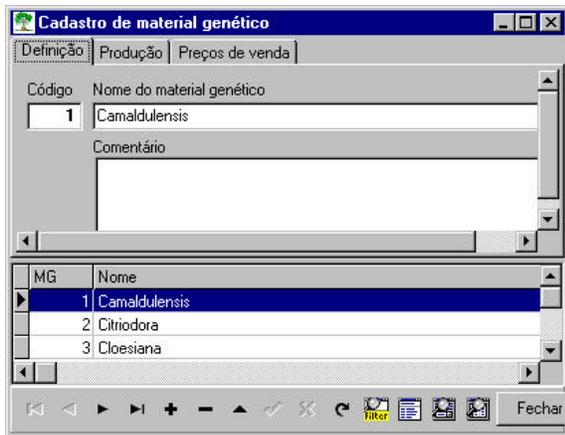


Figura 5 - SisFlor : módulo materiais genéticos.

3.1.1.4. Prognóstico

Este módulo armazena informações de produções ou volumes de madeira para cada material genético. A estrutura apresentada na Figura 6 é o formato de tabelas, com informações de produção (m^3/ha) por unidade de manejo (UM), regime de corte, idade e material genético (MG).

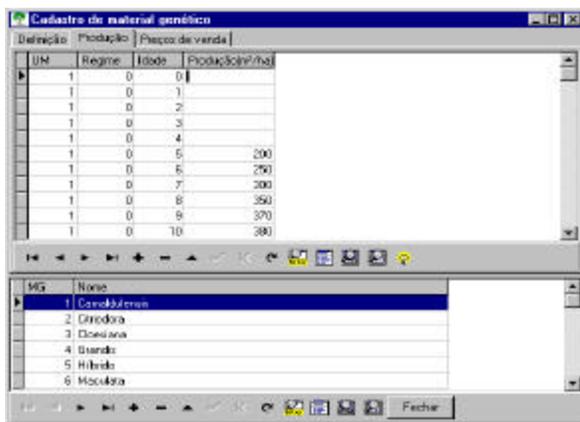


Figura 6– SisFlor: módulo de prognóstico.

3.1.1.5. Custos

Os custos incidentes na atividade florestal foram classificados segundo as categorias: manutenção, implantação, reforma, colheita (exploração), pré-corte, transporte, custo da terra (Figura 7). Cada tabela de custo apresenta uma estrutura

específica. Por exemplo, o custo de manutenção foi definido para cada unidade de manejo, uso da madeira, regime e idade.

UNM	Uso	Regime	Idade	Custo(\$/ha)
131	3	1	5	40,25
132	0	1	5	40,25
132	1	1	5	40,25
132	2	1	5	40,25
132	3	1	5	40,25
133	0	1	5	40,25
133	1	1	5	40,25
133	2	1	5	40,25
133	3	1	5	40,25

Figura 7 - SisFlor: módulo de custos.

3.1.1.6. Destinos da madeira

Informações dos destinos ou locais em que a madeira será entregue pelo fornecedor, e assim encerrando-se todos os seus compromissos podem ser fornecidos. Estes pontos podem representar, por exemplo, diferentes portos, diferentes fábricas, diferentes pátios de estocagem da madeira etc. Além de conter os diferentes destinos da madeira, as distâncias médias de cada local de entrega em relação a cada uma das unidades de manejo podem ser fornecidas (Figura 8). Estas distâncias, dentre outras aplicações, são úteis para avaliação de custos de transporte.

Unidade	Área(ha)	Local de entrega	Distância(km)
A	100,0	LOCAL_1	10
B	200,0	LOCAL_1	20
C	50,0	LOCAL_1	30
D	70,0	LOCAL_1	50
E	90,0	LOCAL_1	25
F	100,0	LOCAL_1	60
G	65,0	LOCAL_1	65
H	45,0	LOCAL_1	70
I	95,0	LOCAL_1	100
J	110,0	LOCAL_1	50

Figura 8 - SisFlor: módulo distâncias das unidades de manejo em relação aos possíveis destinos da madeira.

3.1.1.7. Usos da madeira

A inclusão de usos múltiplos da madeira em planos de manejo florestal é uma prática cada vez mais comum nas empresas florestais. Além de ser uma realidade em muitas empresas, o manejador é frequentemente solicitado a analisar a inclusão de novos usos, avaliando seus impactos no planejamento global da empresa. Portanto, ainda que a empresa não pratique usos múltiplos da madeira, em algum instante ela poderá estar interessada em avaliar a possibilidade de adotá-los como alternativa de manejo, por razões econômicas, estratégicas ou até mesmo social. Um sistema que tenha a flexibilidade de análise destas alternativas é, portanto, fundamental.

O módulo do SisFlor onde são cadastrados potenciais usos da madeira é apresentado na Figura 9. Cada uso possui um código numérico, variando de zero até um número inteiro igual ao número de usos definidos pelo usuário.

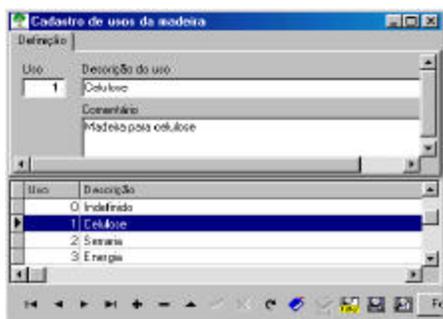


Figura 9 - SisFlor: módulo dos usos potenciais da madeira

3.1.1.7. Preços de venda de terras

As empresas florestais defrontam também com o problema de definir programas de venda ou descarte de terras em função das baixas produtividades do locais, estratégia para reduzir raio médio de transporte, liquidação de investimentos, dentre outros. Assim, o preço de aquisição de terra é um fator importante no estabelecimento do programa de venda de terra, afetando obviamente a análise de fluxo de caixa das alternativas de manejo que simulam a venda de terra. Os preços de venda de terra são definidos no sistema em um módulo específico, conforme Figura 10. No exemplo da tabela de preços em foco, o seu valor é especificado para cada unidade de manejo.

UM	Preço(\$/ha)
1	900
2	900
3	900
4	900

Figura 10 – SisFlor: módulo preços da terra.

2.1.1.8. Preços da madeira

O preço da madeira deve ser fornecido para cada uso da madeira, material genético e estratégias de comercialização (madeira em pé e madeira colhida), conforme esquema da Figura 11.

Uso	Explorado(R\$/m³)	Em pé (R\$/m³)
1	R\$20,00	R\$15,00
2	R\$25,00	R\$20,00

MG	Nome
1	Canaló, Jeneio

Figura 11 – SisFlor: módulo preços da madeira.

3.1.1.9. Outros módulos

Outras informações como taxa de juros, horizonte de planejamento, especificação de clientes são fornecidas em outros módulos especiais do sistema.

3.1.2. Módulo para modelagem

Este módulo constitui-se na “espinha dorsal” do sistema. Além de usar a base de dados do módulo de informações, ele gera novas informações para o sistema. A geração de alternativas de manejo, os resultados da análise econômica

das respectivas alternativas, são algumas dessas novas informações. A flexibilidade na formulação dos modelos de suporte à decisão dependem diretamente deste módulo. Nele são definidos, por exemplo, o intervalo de variações permitidas na idade de colheita, a duração do ciclo de corte, opções por desbaste ou corte raso, escolha do critério de otimização, escolha do modelo de otimização a ser utilizado, avaliações de compra ou venda de terras, dentre outros. Este é também, o módulo que requer maior conhecimento e criatividade do usuário. As principais características e funções deste módulo são apresentadas de forma resumida a seguir.

3.1.2.1. A matriz de manejo

A geração de alternativas de manejo é uma das etapas mais importantes na solução de problemas de planejamento florestal. Como a solução do problema depende dessa tarefa, a geração de um conjunto de alternativas que representem com fidelidade a realidade do problema que se propõe resolver é fundamental. Uma alternativa ou opção de manejo é uma seqüência de ações que ocorrem durante um horizonte temporal. A modelagem de problemas de planejamento florestal pode ser efetuada por duas abordagens distintas de geração de alternativas de manejo: modelo I e modelo II, conforme descritas por JOHNSON e SHEURMAN (1977). A abordagem de modelagem utilizada pelo sistema utiliza apenas o modelo I.

Em função das inúmeras decisões impostas pelo manejador e pela própria natureza dinâmica dos povoamentos florestais, é possível listar uma infinidade de alternativas de manejo para se manejar uma floresta. Por si só, a variável tempo atua como fator de mudança da estrutura do povoamento, alterando, por exemplo, sua idade e produção. Quando aliado a fatores humanos, os seus efeitos sobre a dinâmica do povoamento são ainda mais potencializados. Por exemplo, a intervenção de colheita altera a idade e o regime de corte do povoamento se um corte raso é efetuado. A todo instante, decisões são tomadas e, alteradas pelo fator tempo, há uma constante mudança na estrutura da floresta. Representar esta dinâmica, projetando-a no tempo de forma que diferentes alternativas de manejo

sejam criadas é o desafio maior do manejador. Entretanto, listar todas as alternativas disponíveis para um dado problema requer a utilização de sistemas especiais.

O SisFlor utiliza uma filosofia de geração muito flexível e fácil de manipular. Através de adaptações e melhoramentos da metodologia proposta por TAUBE NETTO (1984), o SisFlor adota uma estrutura de matriz de manejo que confere enorme flexibilidade na geração dessas alternativas.

A matriz de manejo é constituída por um conjunto de estados inter-relacionados (Quadro 5). Um estado é uma variável que pode representar diferentes operações florestais (plantio, reforma, corte raso, desbaste, condução da brotação, combate a formigas, capina etc); diferentes estágios de desenvolvimento da floresta (diferentes idades do povoamento) e diferentes decisões do manejador (compra ou venda de terra, deixar áreas em repouso etc). Exemplos de alguns estados possíveis de um dado contexto florestal são apresentados no Quadro 5. Pode-se observar (Quadro 5) que, além da definição dos possíveis estados, a matriz de manejo descreve a relação de transição entre eles. Conforme definido por TAUBE NETTO (1984), a *transição* de um estado para outro pode ser determinada pela *evolução natural* da floresta. Por exemplo, no Quadro 5, a seqüência de estados 1-2-3-4 é uma seqüência de transição natural da floresta. A transição entre os estados pode ser também *definida* pelo manejador. Por exemplo, de acordo com o mesmo Quadro 5, se a floresta estiver no estado 4, no ano seguinte, pode-se decidir por transferi-la para o estado 05 (mais um ano de crescimento), CR (colheita com reforma no mesmo período), C (colheita sem reforma no mesmo período), CT1 (colheita e condução por talhadia 1), CT2 (colheita e condução por talhadia 2), V (venda ou descarte da área) ou D (desbaste).

As transições possíveis, além de dependerem do estado ocupado, podem depender do período do horizonte de planejamento, do regime de corte vigente etc. Por exemplo, a transição do estado 4 para o estado CT1 só deve ser permitida se no estado 4 o povoamento estiver em regime de alto-fuste. Da mesma forma, a transição do estado 4 para o estado CT2 só deve ser permitida se no estado 4 o povoamento se encontrar em regime de primeira brotação. Esta condição deve ser colocada uma vez que a seqüência natural entre regime de manejo é do alto-fuste

para primeira brotação e desta para segunda brotação. Por outro lado, em alguns casos pode ser desejável proibir vendas de terras após um dado período do horizonte ou mesmo forçar a colheita de povoamentos mais velhos no início do horizonte.

O modelo de matriz de manejo representado no Quadro 5 foi adaptado do trabalho de TAUBE NETTO (1984). O modelo proposto neste trabalho diferencia do modelo proposto por TAUBE NETTO (1984) no sentido de que neste novo modelo não há necessidade de especificar os regimes de manejo e usos da madeira para os respectivos estados da matriz de manejo, ao contrário do modelo anterior, onde os estados da matriz de manejo devem ser especificados a nível de regime de manejo e usos da madeira. As vantagens da não necessidade de detalhar os estados a nível de regime de manejo e uso da madeira é a redução no número de estados necessários para representar um dado problema.

Quadro 5 - Matriz de manejo com os possíveis estados em um contexto florestal²

Estado Atual	Descrição	Estado no período seguinte
1	1 ano de idade	2
2	2 anos de idade	3
3	3 anos de idade	4
4	4 anos de idade	5, D, CR, C, V, CT1, CT2
5	5 anos de idade	6, D, CR, C, V, CT1, CT2
6	6 anos de idade	7, D, CR, C, V, CT1, CT2
7	7 anos de idade	8, D, CR, C, V, CT1, CT2
8	8 anos de idade	9, D, CR, C, V, CT1, CT2
9	9 anos de idade	10, D, CR, C, V, CT1, CT2
10	10 anos de idade	D, CR, C, V, CT1, CT2
CT1	Corte raso seguido por primeira rebrota - talhadia 1	1
CT2	Corte raso seguido por segunda rebrota - talhadia 2	1
CR	Corte raso com reforma no mesmo ano da colheita	1
C	Corte raso sem plantio ou reforma no mesmo ano	R, AGR
D	Desbaste do projeto	5, 6, 7, 8, 9, D, CR, C, V, CT1, CT2
R	Reforma – novo plantio em área colhida	1
AGI	Aguardando Implantação	AGI, I, V
AGC	Projeto aguardando compra	AGC, I

² Adaptado de TAUBE NETTO (1984)

AGR	Área aguardando reforma	R, AGR
I	Implantação: plantio em área com alteração de uso	I
V	Venda ou descarte de terra após o corte	V
VD	Projeto vendido	VD

Uma vez definida a matriz de manejo com todos os possíveis estados e suas relações de transição para um dado problema, o passo seguinte é projetar os estados numa escala temporal igual ao tamanho do horizonte de planejamento (H). Estes estados projetados no tempo criam uma rede de estados com formato do esquema da Figura 12. Os nós da rede dessa Figura 12 representam os limites entre dois estágios (períodos do horizonte de planejamento), sendo o limite de transição entre dois estados. Os arcos, conectando os nós, representam a duração da decisão tomada, que é igual ao tamanho de cada período do horizonte de planejamento. Seguindo a filosofia

de geração de alternativas de manejo via modelo I, onde cada alternativa representa uma seqüência única de decisões, pode -se identificar uma alternativa de manejo como uma rota viável, partindo de um nó inicial (origem) e chegando a um nó final (destino) da rede de opções. Iniciando-se com o estado 1 no primeiro período do horizonte de planejamento, e baseado na estrutura da matriz de estados do Quadro 5, até o sexto período do horizonte de planejamento, tem-se a estrutura de rede da Figura 12. Existem 19 rotas distintas, com cada uma delas representando uma seqüência única de decisões, ou seja, uma alternativa de manejo para o problema. Das 19 alternativas possíveis, apenas 17 seriam viáveis, já que iniciando com o estado 1 indo para o estado 4 e 5 e assumindo que o regime de manejo inicial é de alto-fuste, o estado CT2 (corte e condução por segunda brotação) não seria gerado (linha pontilhada), dada a sua incompatível seqüência natural (um regime de alto-fuste não pode ser transformado diretamente em um regime de segunda brotação). A transição 4-CT2 e 5-CT2 seriam viáveis se o regime no estado 4 e 5 fossem de primeira talhada.

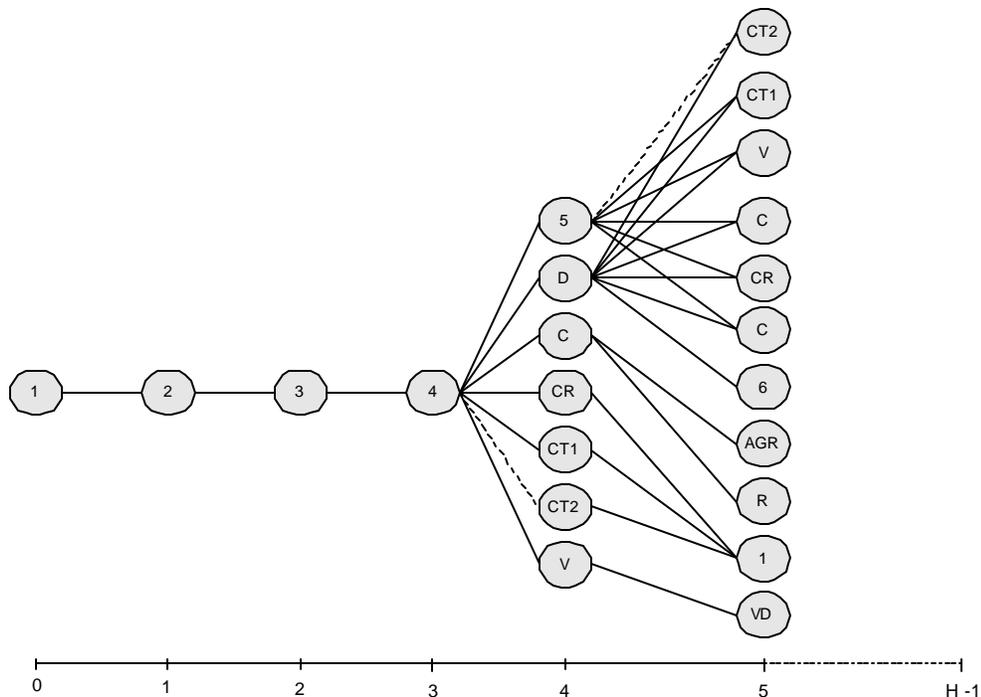


Figura 12 – SisFlor: esquema de uma rede de prescrições de manejo, considerando a matriz de estados do Quadro 5.

A matriz de manejo é definida no sistema por um módulo especial, conforme esquematizado na Figura 13. Para cada estado existe um código alfa-numérico e uma descrição para identificá-lo. Por exemplo, o estado escolhido para representar uma área colhida em um dado período do horizonte de planejamento e deixada para ser reformada no período seguinte foi codificado como "C". A decisão de colher uma área e não reformá-la no mesmo período pressupõe pelo menos duas decisões possíveis no período seguinte do horizonte de planejamento: reforma da área (estado "R") ou deixá-la por mais um período aguardando reforma (estado "AGR"). Para cada estado pode ser escolhido a ocorrência de um ou mais dos seguintes eventos: Plantio ou reforma, produção, venda de terras, conforme consta no canto inferior da tela mostrada na Figura 12. Por exemplo, a seleção do evento produção indica que receitas serão auferidas naquele estado. A produção pode ser efetuada segundo dois tipos de corte: "corte raso" ou "desbaste". A opção escolhida para o estado C foi "corte raso", conforme representado na Figura 12. A produção via corte raso altera também o regime de manejo. Por exemplo, se um povoamento encontra-se em

regime de alto fuste, após o corte sem reforma, seu regime será alterado para regime de primeira brotação (talhadia 1). Para cada estado é definido ainda uma expressão de validação que especifica uma condição em que um determinado estado será gerado. Por exemplo, segundo a Figura 12, a expressão de "validação" ($IDADE < 8$) especifica que o estado C só será gerado se a idade do povoamento for menor do que 8 (anos). A expressão de validação é um artifício utilizado pelo gerador de alternativas de manejo para diminuir o número de prescrições geradas. Critérios técnicos e econômicos podem ser utilizados para esta finalidade. Por exemplo, um estudo preliminar poderia indicar a melhor faixa de idades de colheita para um povoamento, evitando geração de alternativas de manejo com idades de colheita fora dessa faixa.

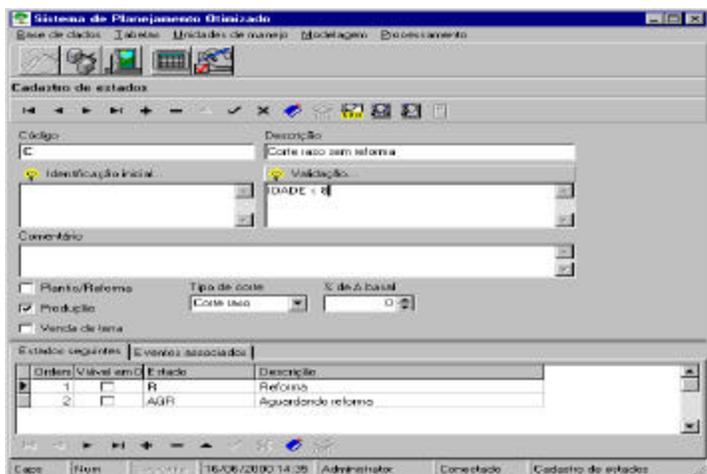


Figura 13 – SisFlor: módulo para definição da matriz de manejo.

Opções de desbastes podem ser consideradas na matriz de manejo. A versão atual do sistema simula desbaste através do controle da área basal. Por exemplo, um estado para representar um desbaste com redução de 30% da área basal foi codificado como "D1", conforme representado na Figura 13. O tipo de corte escolhido para "D1" é, obviamente, "desbaste". Para este tipo de corte deve ser especificado a percentagem de redução da área basal, no exemplo em foco de 30%. Este valor é editado no campo "% de A.basal", conforme representado na Figura 14.

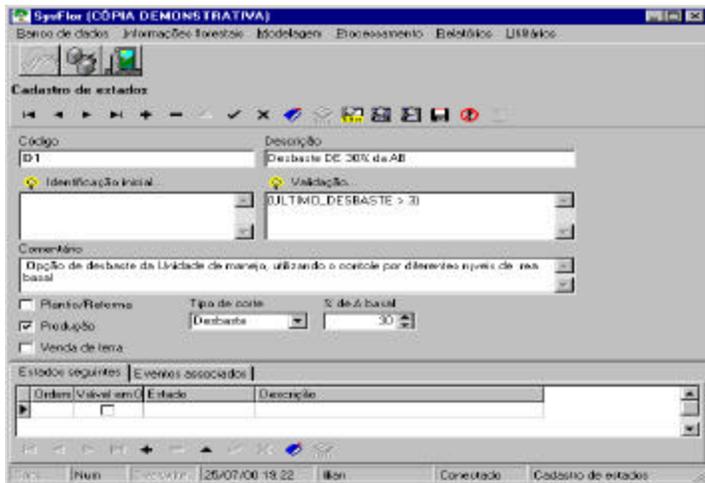


Figura 14 – SisFlor: módulo para definição da matriz de manejo, ilustrando um estado para representar desbaste.

3.1.2.2. Substituição de material genético

No planejamento de médio e longo prazo é comum o manejador deparar-se com problemas de substituição de material genético. Programas de melhoramento florestal estão ocorrendo a todo instante com o objetivo de produzir novos materiais genéticos para substituírem aqueles com baixa produtividade e qualidade indesejável. A substituição de materiais genéticos pode ter impacto em todo processo produtivo. Operações de implantação são alteradas para viabilizar a troca, novos tratamentos culturais são requeridos, alterando a estrutura de custos, alterações na produção podem também ocorrer, afetando a oferta de madeira. A quantificação desses impactos não podem ser dispensadas pelo manejador. A medição desses impactos pode auxiliar em muito na decisão de substituir ou não um determinado material em função do custo-benefício da troca.

O sistema SisFlor possui um módulo especial para simular substituições de material genético. De posse da lista de todos os materiais genéticos cadastrados no módulo "material genético", para simular uma substituição de material genético, o usuário necessita somente definir a condição em que a substituição ocorrerá. Na Figura 15 é mostrado um exemplo onde é definido uma substituição.

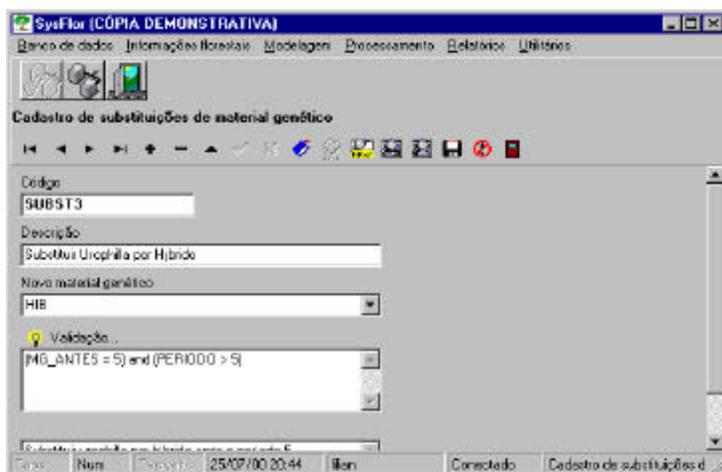


Figura 15 – SisFlor: módulo para substituição de material genético.

Para simular a substituição de um dado material genético, o usuário especifica um código para a substituição, com respectiva descrição da troca. Em seguida, escolhe o novo material genético, disponibilizados em uma lista, e finalmente, especifica as condições para que a substituição ocorra. O exemplo em foco mostra a substituição do material genético *E. urophila*³ (MG_ANTES = 5), a partir do sexto período do horizonte de planejamento (PERÍODO > 5), pelo material genético Híbrido.

3.1.2.3. Modelos de silvicultura

A produção de madeira para usos mais nobres como: serrarias, laminados, compensados, postes etc, requer manejos diferenciados do povoamento. O conjunto de práticas adotadas para produção de madeira para uma determinada finalidade, representa um modelo de silvicultura e resulta em um conjunto de práticas de manejo específica para o povoamento. Por exemplo, desbaste, desrama, diferentes espaçamentos etc são algumas práticas silviculturais que podem ou necessitam ser adotadas para produção de madeira para uso mais nobre como serraria, tendo impactos no fluxo de caixa da empresa. Tais modelos podem ter, ainda, duração temporária ou permanente, podendo apresentar restrições de aplicações. Estes são

³ O código dessa espécie é representado pelo código numérico "5" mostrado na expressão entre parêntese.

apenas alguns dos aspectos que afetam a utilização desses modelos, constituindo um problema bastante complexo para o manejador florestal.

A exemplo da substituição de materiais genéticos, o SisFlor possui uma interface que permite a simulação de substituições dos modelos de silvicultura (Figura16).

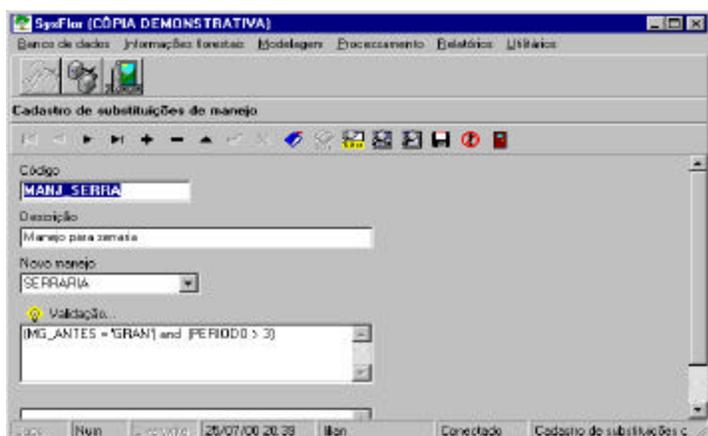


Figura 16 – SisFlor: módulo para substituição de modelos de silvicultura.

3.1.2.4. Gerador de alternativas de manejo

Este módulo é representado por um programa computacional que se encarrega da geração das alternativas de manejo. Baseado nas informações da matriz de manejo, substituição de material genético, modelos de silvicultura e da base de dados do módulo de informações, o programa lista todas as possíveis alternativas de manejo, segundo a filosofia do modelo I (Figura 17).

Observe, por exemplo, a opção ou alternativa de manejo 15, gerada para a unidade de manejo A, com dois anos de idade, no início do horizonte de planejamento (período 0), em regime de segunda brotação (T2). Partindo do estado inicial 02, que representa um povoamento com dois anos de idade, e um horizonte de planejamento de 10 anos, tem-se a seqüência de estados mostradas na Figura 17. No período 6 é prescrito um corte sem reforma imediata da área (estado C), cuja produção poderá ser alocada para dois sortimentos (usos) distintos (Celulose e energia). A opção prescreve uma reforma da área (estado R) no período 7, que nos

períodos seguintes (período 8 e 9) terão, respectivamente 1 e 2 anos de idade. Observe a mesma opção 15 e note que após o corte no período 6, o regime de manejo foi modificado de T2 para T3. A reforma no período 7 restabelece o regime de alto-fuste (AF) através da operação de plantio ou reforma.

OPÇÃO_ID	UNIDADE	PERÍODO	IDADE_DEPOIS	EST_DEPOIS	REGIME_DEPOIS	MANEJO	QJENTE	SORTIMENTO	EST_ANT
14 A		0	1 01	AF	CELULOSE			CR	
14 A		1	2 02	AF	CELULOSE				01
14 A		2	3 03	AF	CELULOSE				02
14 A		3	4 04	AF	CELULOSE				03
14 A		4	5 01	AF	CELULOSE	CU_1	CELULOSE		04
14 A		5	6 05	AF	CELULOSE				01
14 A		6	7 06	AF	CELULOSE				05
14 A		7	8 07	AF	CELULOSE				06
14 A		8	9 02	AF	CELULOSE	CU_1	ENERGIA		07
14 A		9	10 09	AF	CELULOSE				02
15 B		0	2 02	T2	CELULOSE				01
15 B		1	3 03	T2	CELULOSE				02
15 B		2	4 04	T2	CELULOSE				03
15 B		3	5 05	T2	CELULOSE				04
15 B		4	6 06	T2	CELULOSE				05
15 B		5	7 07	T2	CELULOSE				06
15 B		6	0 C	T3	CELULOSE	CU_1	ENERGIA		07
15 B		6	0 C	T3	CELULOSE	CU_1	CELULOSE		07
15 B		7	0 R	AF	CELULOSE				C
15 B		8	1 01	AF	CELULOSE				R
15 B		9	2 02	AF	CELULOSE				01
16 B		0	2 02	T2	CELULOSE				01
16 B		1	3 03	T2	CELULOSE				02
16 B		2	4 04	T2	CELULOSE				03
16 B		3	5 05	T2	CELULOSE				04
16 B		4	6 06	T2	CELULOSE				05
16 B		5	7 07	T2	CELULOSE				06

Figura 17 - SisFlor: alternativas de manejo geradas.

3.1.2.5. Restrições do modelo

Para formulação dos modelos de Pesquisa Operacional é necessário que sejam estabelecidos suas restrições. Dado a infinidade de possibilidades de ocorrências dessas restrições (por exemplo, restrições de área, de produção, de capital, mão-de-obra, financeiras, ambientais etc), é importante que o módulo responsável pela formulação dessas restrições seja bastante flexível. O SisFlor permite uma enorme flexibilidade na formulação dessas restrições de uma forma relativamente simples. Para tal, o usuário necessita apenas de noções básicas sobre a estrutura matemática das restrições de um problema de Programação Linear (PL).

Uma restrição de um problema de PL é uma equação ou inequação de natureza linear do tipo:

$$\sum_{i=1}^n \sum_{j=1}^m a_{ij} X_{ij} \geq b_i \quad (1)$$

em que:

X_{ij} = a variável de decisão;

a_{ij} = coeficiente tecnológico; e

b_i = nível da restrição.

A variável X_{ij} representa uma alternativa de manejo, expressa em unidades de área (ha). Os coeficientes técnicos podem ser, por exemplo, numa restrição de demanda de madeira, os rendimentos volumétricos (m^3/ha). O nível desejado (b_i) de uma dada restrição pode ser, por exemplo, a cota desejada de produção de madeira em um determinado período do horizonte de planejamento. Finalmente, os operadores representam o atendimento mínimos, máximos ou exatos dos níveis de recursos estabelecidos, dependendo se eles são, respectivamente (\geq , \leq , $=$).

A interface do sistema que permite a definição de restrições é muito flexível e fácil de utilizar (Figura 18). Após escolhido um código para identificar a restrição, o usuário deve definir uma máscara (variável auxiliar utilizada para calcular os níveis da restrição) para cada categoria de restrição. O campo agrupamento permite que o usuário defina uma consulta a banco de dados, baseada na linguagem SQL (Search and Query Language). Esta ferramenta propicia uma enorme flexibilidade na formulação das restrições do modelo, uma vez que, permite criar diversos níveis de consulta. Por exemplo, se o usuário desejasse criar uma restrição de produção por uso da madeira em cada período (ano) do horizonte de planejamento, poderia escolher uma consulta onde fosse estabelecido um agrupamento das restrições por "Uso,ano" conforme representado na Figura 18. Para cada categoria de restrição deve ser definido o seu coeficiente tecnológico. Na restrição de demanda de madeira, por exemplo, este coeficiente é a própria produção por unidade de área. Em seguida, o usuário deve definir o operador da restrição, que no caso da restrição de demanda máxima por uso e ano é do tipo "<". A opção "filtro" mostrado no esquema da Figura 18 mostra o resultado da consulta para cada categoria de restrição, conforme representado na Figura 19.

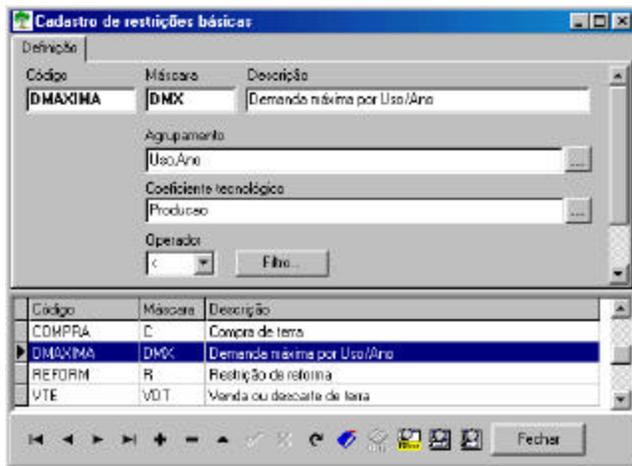


Figura 18 - SisFlor: módulo para definição das restrições do problema.

Por exemplo, para restrição de demanda por uso e por ano poderíamos ter um resultado do tipo mostrado na Figura 19 onde são listados as categorias de uso, os anos em que ocorrem, as produções e o código da opção de manejo. Esta consulta é utilizada, pelo módulo gerador de modelo para criação de restrições.



Figura 19 – SisFlor: resultado de uma consulta definida no módulo formulador de restrições.

A última etapa do processo de definição de uma restrição é a edição dos seus níveis desejados. A rotina cria uma tabela, listando as variáveis definidas pelo agrupamento do módulo de criação de restrições, com os respectivos níveis da restrição especificada (Figura 20). De posse da tabela com as informações sobre a restrição, o usuário necessita apenas estabelecer o nível desejado de cada restrição. No exemplo da Figura 20, estes níveis foram iguais a 45.000, o que significa que a

demanda máxima por uso em cada ano (período) do horizonte de planejamento deverá ser de 45.000 m³ de madeira .



Código	Máscara	Descrição
AGI	AGI	Áreas novas aguardando implantação
COMPRA	C	Compra de terra
DMAXIMA	DMX	Demanda máxima por Uso/Ano
REFORM	R	Restrição de reforma

Índice	Ano	Nível
DMX01	0	45000
DMX11	1	45000
DMX21	2	45000
DMX31	3	45000
DMX41	4	45000
DMX51	5	45000
DMX61	6	45000
DMX71	7	45000
DMX81	8	45000
DMX91	9	45000

Figura 20 – SisFlor: módulo para edição dos níveis das restrições do problema.

3.1.3. Módulo gerador de modelos

Este módulo é composto por um programa computacional para gerar o modelo matemático. Nele, todas as informações contidas em arquivos ou tabelas são unidas para formar uma base única de dados, gerando uma matriz básica para construção do modelo.

A versão atual do SisFlor permite formular problemas de planejamento florestal utilizando-se de modelos PL e PI. Na Figura 21 é apresentado a janela com as informações requeridas para a formulação de um problema utilizando-se o modelo de PL. Os critérios de otimização disponíveis para maximização ou minimização são: receita líquida, receita bruta, produtividade física e custo total. Pode-se efetuar a flexibilização das restrições de áreas através da escolha da opção "Menor (<)" que atribui o operador "<" para as restrições de área do modelo de PL. Um arquivo de saída é gerado após ativar a opção "Gerar modelo" no canto inferior da tela (Figura 21). A solução do modelo gerado pode ser processada através da opção "Obter resultado" do canto inferior direito da janela (Figura 21).

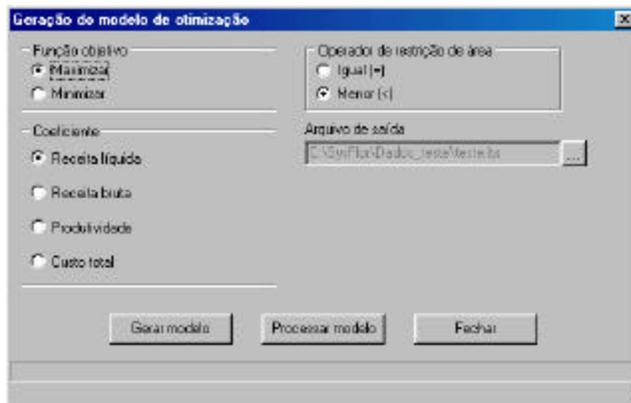


Figura 21 – SisFlor: módulo para geração de modelos de PL.

3.1.4. Módulo de otimização

O sistema SisFlor utiliza os algoritmos *Simplex* e *branch and bound* para solução de problemas de PL e PI, respectivamente, implementados no pacote comercial LINDO. Procedimentos heurísticos baseados em Algoritmos Genéticos, Busca Tabu e *Simulated Annealing* foram utilizados como alternativas para solução dos problemas de PI. Estas heurísticas são objetos de estudo dos capítulos seguintes.

3.1..5. Módulo de relatórios

O sistema disponibiliza diversas alternativas de relatórios. Por exemplo, relatórios com informações relativas ao plano ótimo com informações sobre áreas de cada alternativa alocada nas respectivas unidades de manejo, seqüência de decisão (estados), idades e regime de manejo (Figura 22).

Copener Florestal Ltda
 Divisão de Planejamento
 Plano de manejo otimizado
 Valor da função objetivo: 0,0

Ano	U.M.	Área (ha)	Mat. Genêr.	Esp.	Ass.	Região	Idade (anos)	Espécie	Produção (m³/ha)
27	79,20	Testeoma	0			Alfata	13	30	
27	79,20	Testeoma	0			3 Alfata	14	30	
27	79,20	Testeoma	2			2 Alfata	13	CITD	200M
27	79,20	Testeoma	0			3 TI	-	81	
27	79,20	Testeoma	0			4 TI	3	82	
27	79,20	Testeoma	0			5 TI	2	83	
27	79,20	Testeoma	0			6 TI	3	84	
27	79,20	Testeoma	0			7 TI	4	85	
27	79,20	Testeoma	0			8 TI	5	86	
27	79,20	Alfata	2			3 TI	8	CEB	300M
28	300,00	Testeoma	0			8 Alfata	13	30	
28	300,00	Testeoma	0			3 Alfata	14	30	
28	300,00	Testeoma	0			3 Alfata	15	7795	120M

Página 11 de 11

Figura 22 – SisFlor: relatório do plano de manejo otimizado.

Os principais custos e receitas periódicos, discriminados; em valor do ano corrente (não descontados) para o plano ótimo é uma outra opção de relatório. Na Figura 23 é apresentado o esquema de relatório para os custos periódicos.

Copener Florestal Ltda
 Divisão de Planejamento
 Custos anuais da planta de manejo otimizado em \$ (unidades monetárias)

Ano	Implantação	Receita	Manutenção	Explicação	Transporte	Aluguel Terço	Total
0	20022800,00	2718280,00	5347580,00	25067467,57	2122,00	-4026378,50	63262541,25
1	0,00	6167281,63	7963689,62	3256510,50	714,00	-4026378,50	29444252,64
2	20022400,00	2638571,24	6825486,02	35462117,59	7725,00	-5226378,50	113401289,95
3	0,00	4913385,06	9224263,40	6046722,00	601,00	-5226378,50	28367691,14
4	0,00	-402940,06	9361888,36	6271343,67	934,00	-5226378,50	17442262,40
5	0,00	933932,04	9949205,01	13394039,73	985,00	-5226378,50	29366233,98
6	0,00	6117284,46	5118501,40	15577669,10	1041,00	-5242794,50	32146630,61
7	0,00	2744083,26	5468912,00	65572692,21	1144,00	-5226404,50	73162267,17
8	0,00	0,00	5112203,00	92930670,00	2046,00	-5226404,50	28073421,16
9	0,00	-4993414,18	-4994488,30	100196895,30	7095,00	-5124794,01	121164270,29

Página 1 de 1

Figura 23 – SisFlor: relatório com os principais custos periódicos do plano de manejo.

Outros relatórios como níveis ótimos das restrições especificadas no modelo como: áreas vendidas ou descartadas, áreas compradas, volume periódicos de madeira com respectivos usos, áreas reformadas e implantadas periodicamente, áreas de corte anuais etc estão disponíveis em outras opções de relatório.

3.2. Solução do problema proposto

O problema apresentado no item 2.2 foi formulado por meio dos modelos de PL e PI utilizando-se o sistema de planejamento SisFlor. A formulação do problema descrito no estudo de caso resultou num total de 215 variáveis de decisão e 49 restrições, sendo 19 restrições de área, 10 restrições de demanda mínima, 10 restrições de demanda máxima e 10 restrições de reforma. Para o modelo de PI, uma restrição adicional foi necessária para declarar as variáveis de decisão como inteiras. As soluções do problema para os modelos de PL e PI foram obtidas, respectivamente, através dos algoritmos *simplex e branch and bound* implementados no software Extended LINDO/PC, versão 6.01 (1997).

As soluções ótimas para os modelos PL e PI são apresentadas no Quadro 6. Estas soluções representam dois planos alternativos de manejo para o problema proposto. O valor ótimo do plano resultante com otimização via PL foi de \$ 4.387.365,00, com as recomendações de alternativas de manejo para cada unidade de manejo mostradas no Quadro 6. Pode-se observar que em algumas unidades de manejo foram alocadas mais de uma alternativa de manejo, o que é já era esperado, uma vez que a propriedade de continuidade das variáveis de decisão no modelo de PL permite tal ocorrência. Por exemplo, para a unidade de manejo 1, com uma área total de 50 ha foram alocadas duas alternativas de manejo (X4 e X6) com respectivas áreas de 23,96 e 26,04 ha. A alocação de mais de uma alternativa de manejo para uma mesma unidade de manejo pode inviabilizar a implementação do plano ótimo toda vez que uma dada unidade de manejo for considerada indivisível para efeito de manejo. Os talhões florestais ou as unidades mínimas de manejo são considerados indivisíveis em diversas operações como: colheita, plantio, reforma, tratos culturais etc. Portanto, a solução apresentada para o problema através do modelo de PL não é uma solução operacional segundo esta ótica.

Existem duas soluções possíveis para o problema de fracionamento das unidades mínimas de manejo, as quais consistem no arredondamento das soluções ótimas e a utilização de modelos adequados para solucionar o problema. A opção por arredondamento foi avaliada por SILVA (2001). A conclusão deste trabalho foi de que o uso deste artifício resultou, na maioria das vezes, em soluções inviáveis para o problema. GOLDBARG e LUNA (2000) ressaltaram que a utilização de

arredondamento, além de resultar em inviabilidades das soluções resultantes, pode sacrificar todo o esforço de modelagem devido à má qualidade das soluções obtidas com este procedimento. A Segunda abordagem, é formular e resolver o problema utilizando modelos de PI, conforme apresentado no Quadro 6. Observe que a solução por esta abordagem permitiu a alocação de uma única alternativa de manejo em cada unidade. O valor ótimo desse plano alternativo foi de \$ 4.342.088,00, uma redução de \$ 45.277,00 (1,03%) em relação à solução via PL. Esta redução pode ser interpretada como a penalidade imposta ao se impor a restrição de inteireza nas variáveis de decisão, ou seja, permitir a alocação de apenas uma alternativa de manejo em cada unidade. Além da mudança no valor da função objetivo, a formulação via PI altera as alternativas de manejo (variáveis de decisão) do plano ótimo, conforme pode ser observado no Quadro 6.

Os valores da produção volumétrica em cada período do horizonte de planejamento para os dois planos alternativos obtidos através da solução via PL e PI são mostrados no gráfico da Figura 24. Note que em todos períodos, a produção encontra-se entre os limites mínimos e máximos estabelecido de 30.000,00 m³ e 45.000,00 m³, respectivamente. Totalizando a produção dos dois planos em todos os períodos, obtém-se 407.092,00 m³ e 403.805,00 m³ para as duas formulações (PL e PI), respectivamente, uma diferença superior de 3.287 m³ (0,81%) em favor do modelo de PL.

Quadro 6 - Solução ótima do problema formulado via Programação Linear e Programação Inteira, objetivando a maximização do valor presente líquido sujeitos às restrições de área, demandas mínimas e máximas e áreas de reforma máximas, periodicamente.

Unidades de manejo	Área da Unidade de Manejo (ha)	Solução via Programação Linear		Solução via Programação Inteira		
		Alternativa de manejo	Área (ha)	Alternativa de manejo	Valor	Área (ha)

1	50,00	X_4	23,96	X_6	1	50,00
1		X_6	26,04			
2	20,00	X_{10}	20,00	X_{12}	1	20,00
3	25,00	X_{17}	25,00	X_{17}	1	25,00
4	54,00	X_{26}	54,00	X_{26}	1	54,00
5	30,00	X_{41}	30,00	X_{41}	1	30,00
6	45,00	X_{56}	2,60	X_{66}	1	45,00
6		X_{66}	42,40			
7	42,00	X_{81}	42,00	X_{81}	1	42,00
8	32,00	X_{83}	32,00	X_{83}	1	32,00
9	26,00	X_{101}	26,00	X_{101}	1	26,00
10	41,00	X_{103}	41,00	X_{102}	1	41,00
11	45,00	X_{106}	45,00	X_{107}	1	45,00
12	49,00	X_{111}	47,15	X_{111}	1	49,00
12		X_{113}	1,85			
13	52,00	X_{117}	52,00	X_{117}	1	52,00
14	57,00	X_{126}	14,99	X_{131}	1	57,00
14		X_{131}	42,01			
15	60,00	X_{141}	50,09	X_{151}	1	60,00
15		X_{143}	1,44			
15		X_{151}	8,47			
16	55,00	X_{178}	23,33	X_{180}	1	55,00
16		X_{180}	31,67			
17	35,00	X_{195}	35,00	X_{188}	1	35,00
18	52,00	X_{197}	2,45	X_{200}	1	52,00
18		X_{202}	49,55			
19	39,00	X_{215}	39,00	X_{215}	1	39,00

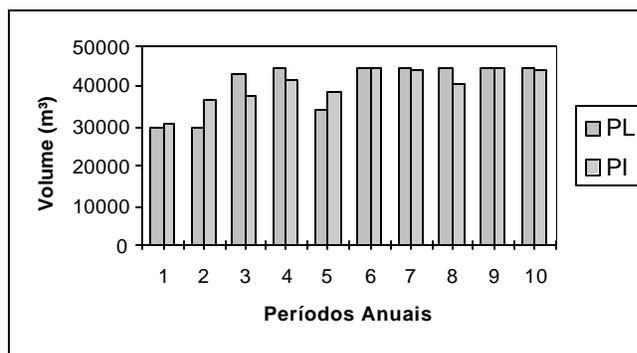


Figura 24 – Volume periódico de madeira produzido em função das formulações via PL e PI.

A solução de problemas de planejamento florestal utilizando modelos de PI seria uma ótima alternativa, não fosse o alto custo computacional para obtenção das soluções ótimas. Para este pequeno exemplo, o tempo de processamento da solução via PI (240 segundos) foi cerca de 120 vezes maior do que o tempo de processamento solução via modelo de PL (2 segundos). De acordo com IGNÍZIO e CAVALIER (1994), os algoritmos exatos (*branch and bound*, *implicit enumeration* e *cutting plane*) são eficiente somente em problemas de tamanho moderado. A maioria dos problemas reais de planejamento florestal são problemas de grande porte, portanto, não factíveis de solução por tais algoritmos. O trabalho de BOSTON e BETTINGER (1999) enfatizou esta limitação. Estes autores, trabalhando com problemas que envolveu pequeno número de variáveis (entre 3.000 e 5.000), solucionados pelo algoritmo *branch and bound* implementado no software CEPLEX constataram, em alguns casos, que o algoritmo não convergiu para a solução ótima após 30 horas de processamento em um microcomputador Pentium 200 Mhz. Na maioria dos casos, a memória do hardware era exaurida. Rodrigues (1994), citado por NOBRE (1999) também encontrou problemas de convergência do algoritmo *branch and bound* para problemas com número reduzidos de variáveis. O autor trabalhando com ótimos globais inteiros para viabilizar a análise de eficácia da Heurística da Razão R, constatou em alguns casos, a não convergência do algoritmo em problemas com aproximadamente de 650 variáveis, antes de esgotar a memória de um microcomputador Pentium 100 Mhz.

A limitação prática dos algoritmos exatos para solução dos modelos de PI e as vantagens comparativas das técnicas heurísticas levaram ao desenvolvimento de inúmeras pesquisas visando a utilização dessas técnicas para resolver problemas de planejamento florestal. Apesar de não garantirem a solução ótima para o problema, as heurísticas alcançam soluções boas (próximas do ótimo) com um tempo computacional muito inferior. Os capítulos seguintes trazem detalhes e aplicações de algumas técnicas heurísticas.

4. CONCLUSÕES

Com base no desenvolvimento e resultados apresentados neste capítulo, pode-se concluir que:

- Além das propriedades naturais dos sistemas de suporte à decisão, o sistema desenvolvido neste trabalho (SisFlor) contribui para melhoria do processo decisório no planejamento de florestas equiâneas, acrescentado as vantagens de permitir a formulação e solução de questões imprescindíveis para o manejo dessas florestas, muitas delas ainda não abordadas por outros sistemas de planejamento;
- O SisFlor diferencia dos principais sistemas de suporte à decisão já desenvolvidos nos seguintes aspectos: formulação e solução de problemas relacionados ao uso múltiplo da floresta, desbaste de povoamentos, programa de substituição e avaliação de materiais genéticos e modelos de silvicultura;
- O módulo gerador de alternativas de manejo, além de permitir flexibilidade na formulação de problemas de planejamento florestal, propicia redução no tamanho da matriz de manejo em consequência do controle sobre as variáveis regime de manejo e uso da madeira;
- A rotina geradora de alternativas de manejo do SisFlor utiliza expressões de validação de estados para evitar a geração de alternativas de manejo indesejáveis, evitando, com isto a explosão do número de variáveis geradas e consequentemente reduzindo o esforço computacional requerido;
- A flexibilidade e facilidade de simulações através de substituições de materiais genéticos e modelos de silvicultura, são outras vantagens do SisFlor, contribuindo para ampliar o leque de aplicações do sistema no manejo florestal;
- Apesar dos avanços conseguidos nesse trabalho e seus objetivos sendo alcançados, algumas melhorias no sistema necessitam ser efetuadas. Melhorias na interface para entrada de dados e relatórios, integração com outros sistemas, em especial os sistemas de cadastro, inventário e sistemas de informação geográfica, são algumas delas. Tais melhorias são fundamentais para facilitar a

utilização do SisFlor nas empresas florestais e serão efetuadas em sua segunda versão.

- O estudo de caso apresentado neste capítulo ressaltou as principais limitações dos modelos de Programação Linear (PL) e Programação Inteira (PI) para solução de alguns problemas de planejamento florestal. Enquanto algoritmo *simplex* é altamente eficiente para solução de problemas de PL de grande porte, apresenta restrições de aplicações em problemas que requerem o uso de variáveis inteiras. Por outro lado, o algoritmo *branch and bound* para solução de problema de PI é eficiente somente em problemas de pequeno porte, o que não é o caso da maioria dos problemas reais de planejamento florestal que, em geral, envolvem centenas a milhares de variáveis de decisão.

5. REFERÊNCIAS BIBLIOGRÁFICAS

BOSTON, K.; BETTINGER, P. An Analysis of Monte Carlo Integer Programming, Simulated Annealing, and Tabu Search Heuristics for Solving Spatial Harvest Scheduling Problems. *Forest Science, Washington*, 45(2):292–301, 1999.

IGNIZIO, J.P.; CAVALIER.T.M. **Linear Programming.** Englewood Cliffs: Prentice Hall, 1994. 666p.

IVERSON, D.C. ALSTON, R.M. **The genesis of FORPLAN a historical and analytical review of Forest Service planning models.** USDA Forest Service General Technical Report, INT-214: p1-25, 1986.

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

JOHNSON, K.N.; JONES, D.; DANIEL, B. **A user's guide to multiple use sustained yield scheduling calculation (MUSYC).** Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Timber Management; 1979. 242 p.

NAVON, D.I. **Timber RAM ... a long-range planing method for comercial timber lands under multiple-use management.** Berkeley, CA: U.S. Department of Agriculture, Forest Service, Pacific Southwest Forest and Range Experiment Station, 1971. 22p. (USDA. Forest Service, Res. paper PNW - 70)

NOBRE, S.R. **A heurística da Razão-R aplicada a problemas de gestão florestal.** Lavras, MG: UFLA, 1999. 128p. Dissertação (Mestrado em Administração Rural) - Universidade Federal de Lavras, 1999.

- RODRIGUES, F.L. **Regulação de florestas equiâneas utilizando programação linear**. Viçosa, MG: UFV, 1997. 109p. Dissertação (Mestrado em Manejo Florestal) - Universidade Federal de Viçosa, 1997.
- RODRIGUEZ, L.C.E.; BORGES, J.G. Técnicas matemática para determinação de níveis sustentáveis de produção florestal: um exemplo em eucalipto. **Revista Florestal**, Lisboa, 12(1), p83-92, 1999.
- SILVA, G.F. **Problemas no uso de programação matemática e simulação em regulação florestal**. Viçosa, MG: UFV, 2001. 89p. Tese (Doutorado em Manejo Florestal) - Universidade Federal de Viçosa, 2001.
- SILVA, M.L. **Influência do diâmetro no custo de corte e na rotação de povoamentos de eucalipto**. Viçosa, MG: UFV, 1991. 71p. Dissertação (Mestrado em Economia Florestal) - Universidade Federal de Viçosa, 1991.
- STAIR, R. **Princípios de sistemas de informação**. Rio de Janeiro. Livros Técnicos e Científicos 1998. 432p.
- TAUBE NETTO, M. Um modelo de programação linear para planejamento de florestas de eucalipto. **Pesquisa Operacional**, Campinas, 4(1), p.19-39, 1984.
- TURBAN, E; ARONSON, J.E. **Decision support systems and intelligent systems**. New Jersey: Prentice-Hall, 1998. 890p.

CAPÍTULO 3

PRINCÍPIOS E APLICAÇÕES DE ABORDAGENS HEURÍSTICAS PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL

RESUMO - As mudanças de paradigmas no gerenciamento florestal têm sido traduzidas pela necessidade de novas abordagens para a solução de problemas neste setor. Modelos e algoritmos anteriormente consagrados têm sido questionados quanto a sua adequação para resolver os novos problemas. Restrições operacionais, restrições ambientais, roteamento de veículos no transporte florestal, dentre outras, são algumas questões que, além de importantes para o tomador de decisão, conferem enormes complexidades aos modelos necessários para formulá-las. Problemas que envolvem estas questões requerem formulação através de modelos de programação inteira com variáveis discretas, em geral, do tipo 0-1. Apesar da facilidade de formulação, os problemas com variáveis discretas são de natureza combinatorial de difícil solução pelos instrumentos computacionais da atualidade. Os algoritmos enumerativos de solução, dentre eles, o algoritmo *branch and bound*, são eficientes apenas em problemas de pequeno porte, o que não é o caso dos problemas reais encontrados no gerenciamento florestal. Nos últimos anos surgiram um conjunto significativo de técnicas e algoritmos computacionais eficientes, mas que não garantem a solução ótima desses problemas. Esses algoritmos foram denominados de *heurísticos* ou *aproximativos*. As heurísticas tidas como mais promissoras para resolver problemas combinatoriais em diversas áreas pertencem às categorias das chamadas metaheurísticas e incluem: *Simulated Annealing* (SA), Algoritmo Genético (AG) e a Busca Tabu (BT). O objetivo desse trabalho foi apresentar os princípios básicos das

**metaheurísticas AG, BT e SA, reportando algumas aplicações no
gerenciamento florestal.**

Palavras-chave: Gerenciamento Florestal, heurísticas, Algoritmos Genéticos, Busca Tabu, *Simulated Annealing*.

**PRINCIPLES AND APPLICATIONS OF HEURISTIC APPROACHES TO
SOLVER FOREST PLANNING PROBLEMS**

ABSTRACT - The paradigm changes in forest management have been explained by the need of new approaches to the solution of problems in this sector. Previously acknowledged algorithms and models have been questioned in relation to their adequacy to solve new problems. Operational and environmental constrains, vehicle routing in forest transport, among others, are some issues that besides being important for the decision maker, present great complexities to the models needed to formulate them. Problems involving these subjects demand formulation via entire programming models with discrete variables, usually of the type 0-1. In spite of the easiness of formulation, the problems with discrete variables are of combinatorial nature of difficult solution by the current computational instruments. The enumerative algorithms of solution, including *branch and bound*, are efficient only for small problems, being not the case of the real problems found in forest management. In the latest years, a significant group of techniques and efficient computational algorithms have arisen, but that do not guarantee the optimum solution of those problems. Those algorithms were designated as heuristic. The heuristics known as more promising to solve combinatorial problems in several areas fall into the categories of metaheuristics and include: Simulated Annealing (SA), Genetic Algorithm (GA) and Tabu Search (TS). The objective of this work was to present the basic principles of the metaheuristics GA, TS and SA, reporting some applications to forest management.

Key-words: Forest Management, Heuristic, Genetic Algorithm, Tabu Search and Simulated Annealing

1. INTRODUÇÃO

Grande parte dos problemas de otimização combinatória apresentam enormes dificuldades de serem resolvidos eficientemente pelos algoritmos exatos tradicionais como *branch and bound*, *implicit enumeration* e *cutting plane*. de acordo com IGNIZIO e CAVALLER (1994), as dificuldades decorrem da explosão combinatória dos métodos enumerativos. Os problemas de planejamento florestal que requerem formulação discreta, através de variáveis inteira 0-1 são problemas combinatoriais. Exemplos comuns são aqueles que demandam restrições de singularidade para evitar que mais de um regime de manejo seja adotado em cada unidade de colheita, restrições espaciais, por exemplo, para evitar que áreas adjacentes sejam colhidas simultaneamente, roteamento de veículos no transporte florestal, dentre outras.

Nos últimos anos surgiram um conjunto significativo de técnicas e algoritmos computacionais eficientes, mas que não garantem a solução ótima desses problemas. Esses algoritmos foram denominados de *heurísticos* ou *aproximativos* (GOLDBARG e LUNA, 2000). As heurísticas mais promissoras para resolver problemas combinatoriais em diversas áreas, incluem: *Simulated Annealing* (KIRKPATRICK, 1983), Algoritmos Genéticos (HOLLAND, 1975) e Busca Tabu (GLOVER 1977). Entretanto, somente recentemente alguns trabalhos foram desenvolvidos com tais técnicas para resolver problemas clássicos de planejamento florestal de natureza combinatorial. Exemplos de aplicações de Busca Tabu e *Simulated Annealing* podem ser encontrados em LOCKWOOD e MORE (1993), LAROZE e GLEBER (1997), LAROZE (1999) e BOSTON e BETTINGER (1999). Considerando as inúmeras aplicações com sucesso dos algoritmos genéticos em outras áreas, acredita-se que estes possam, igualmente, serem aplicados em problemas de planejamento florestal. Além disso, apesar de já existirem trabalhos envolvendo aplicações de Busca Tabu e *Simulated Annealing*, há ainda uma deficiência de estudos visando consolidar seus usos em problemas de planejamento florestal. Diante do exposto, este capítulo teve os seguintes objetivos:

- descrever os princípios básicos da Busca Tabu, Algoritmos Genéticos e *Simulated Annealing*;
- Apresentar algumas aplicações dessas técnicas em problemas de planejamento florestal.

Por se tratar de uma revisão bibliográfica, optou-se por apresentar os resultados da mesma com a separação de alguns temas principais.

2. HEURÍSTICAS

Segundo GOLDBARG e LUNA (2000), o termo heurística é derivado do grego *heriskein*, que significa descobrir ou achar. Segundo os mesmos autores, o significado da palavra em Pesquisa Operacional vai um pouco além de sua raiz etimológica e define uma heurística como *“um método de busca de soluções em que não existe qualquer garantia de sucesso”*. O sucesso do método pode ser expresso quantitativa ou qualitativamente. Em um problema de otimização o sucesso pode ser representado pela obtenção da solução ótima. Alguns autores entendem que a possibilidade de fracasso seja extensiva até para a obtenção de uma solução viável (GOLDBARG e LUNA, 2000). Entretanto, GOLDBARG e LUNA (2000), afirmam não comungarem da idéia e aprofundam a definição da seguinte forma: *“uma heurística é uma técnica que busca alcançar uma boa solução utilizando um esforço computacional considerado razoável, sendo capaz de garantir a viabilidade ou a otimalidade da solução encontrada ou ainda, em muitos casos, ambas, especialmente nas ocasiões em que essa busca partir de uma solução viável e próxima ao ótimo”*.

Segundo Zanakis *et al.*(1981), citado por IGNACIO *et al.* (2000) uma heurística pode ser definida como *“um procedimento simples e rápido, geralmente baseada no sentido comum e se supõe que ofereça uma boa solução (não necessariamente a ótima) no tratamento de problemas complexos”*.

Murray e Church (1995a), citados por MURRAY (1999), generalizaram o processo de solução das heurísticas busca aleatória, *interchange*, *simulated annealing* e busca tabu utilizando o esquema da Figura 1. Apesar do esquema ter

sido desenvolvido para ilustrar o processo geral de solução daquelas heurísticas, ele se aplica a maioria das outras técnicas heurísticas. De acordo com a Figura 1, o processo de busca inicia com a geração de uma solução inicial viável, gerada aleatoriamente ou baseada em algoritmos de geração apropriados. Um critério de parada é então utilizado para testar se a busca continua ou não. Se o critério é atendido, o processo é interrompido; caso contrário um processo para modificar a solução corrente é empregado para identificar uma nova solução viável. Se a solução viável alternativa é encontrada, então a nova solução é aceita. Caso contrário, a nova solução é eliminada, e o processo continua com a solução não modificada. Novamente, o critério de parada é testado para decidir se o processo será interrompido ou seguirá com a busca. Segundo MURRAY (1999), as principais diferenças entre as técnicas heurísticas estão na forma de modificação da solução corrente e no critério de avaliação empregado para determinar se a nova solução é aceita ou não. Outra observação em relação ao processo de busca ilustrado na Figura 1, refere-se à viabilidade da solução. Devido à dificuldade em identificar soluções viáveis em muitos problemas, é comum a alteração dessa regra para aceitar soluções inviáveis, porém, aplicando-se a elas uma penalidade por violar alguma restrição do problema, conforme encontrado em BOSTON e BETTINGUER (1999). Este método, além de melhorar a performance da heurística, ao dispensar o teste de viabilidade da solução, pode funcionar como um artifício para direcionar busca para diferentes regiões do espaço de solução.

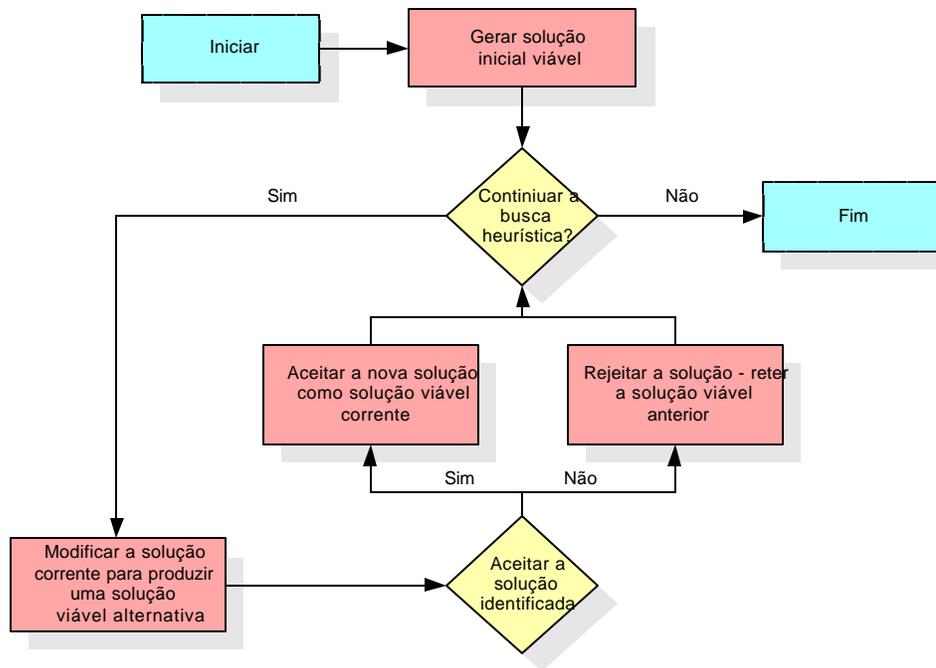


Figura 1 - Processo geral de solução das heurísticas.

Em geral, as heurísticas requerem decisões em relação aos seguintes elementos (YOUSSEF et al, 2001):

- um mecanismo para geração da solução inicial de onde a busca iterativa irá partir;
- uma função de avaliação para avaliar as soluções encontradas durante a busca;
- operadores de perturbação (movimentos) para criar novas soluções a partir de uma dada solução corrente;
- escolha dos parâmetros do algoritmo; e
- critério de parada.

Para IGNÍZIO E CAVALIER (1994), as heurísticas são as abordagens mais promissoras para solução de problemas de Programação Inteira. Segundo estes autores, enquanto os algoritmos exatos para solução de problemas de Programação Inteira garantem uma solução ótima para alguns problemas, as heurísticas não podem provar a otimalidade das suas soluções, mas oferecem, entretanto, soluções aceitáveis, inclusive para problemas complexos e de grande porte, com baixo custo computacional.

Segundo GOLDBARG e LUNA (2000), os primeiros métodos heurísticos eram associados a estratégias de enumeração incompleta, solução parcial ou relaxações e dentre muitas classificações possíveis para as heurísticas, sugeriram a apresentada na Figura 2.

A Figura 2 resume as várias abordagens conhecidas para os procedimentos heurísticos de solução. As heurísticas da linha clássica exploram, de modo casual, a estrutura do problema sem definir claramente uma estratégia universal de solução (GOLDBARG e LUNA, 2000). As primeiras heurísticas relatadas na literatura eram destinadas à solução de problemas específicos e não eram, via de regra, passíveis de serem utilizadas em outros problemas; as heurísticas clássicas de roteamento são um bom exemplo disso (GOLDBARG e LUNA, 2000). Entretanto, nos últimos 30 anos, pesquisadores da área de Pesquisa Operacional adaptaram idéias de outras áreas no desenvolvimento de *metaheurísticas* (SOUZA e MOCCELLIN, 2000).

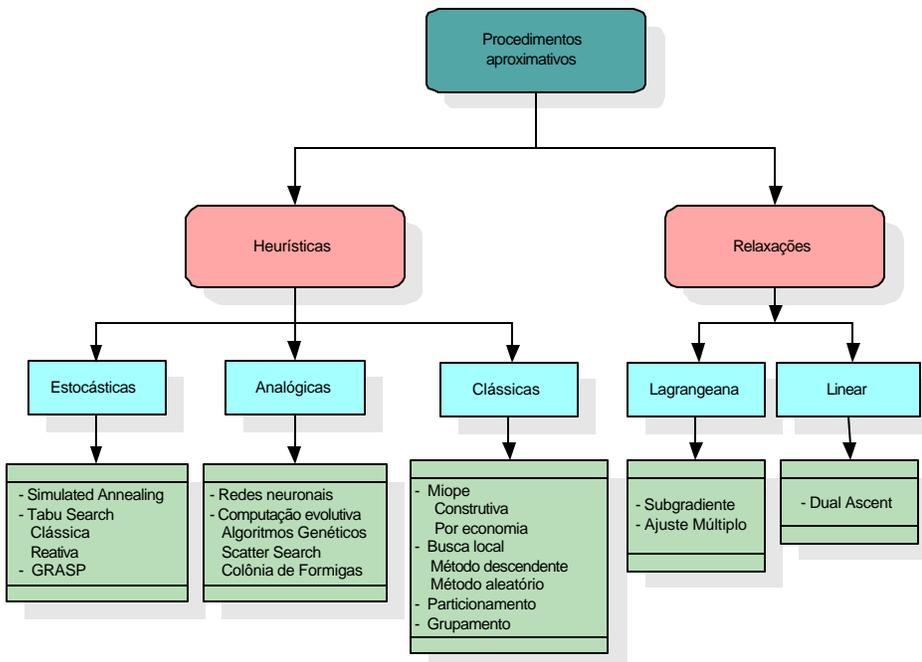


Figura 2: Classificação das abordagens aproximativas (GOLDBARG e LUNA, 2000).

Segundo GOLDBARG e LUNA (2000), o termo metaheurística é reservado para um grupo especial das chamadas heurísticas estocásticas e analógicas. As metaheurísticas são técnicas que, superpondo-se a métodos heurísticos de busca em vizinhança, guiam a busca com vistas à superação da

otimalidade local e à obtenção de soluções de qualidade superior (SOUZA e MOCCELLIN, 2000). Dentre as mais populares, inclui-se *Simulated Annealing* (KIRKPARTRICK, 1983), Algoritmos Genéticos (HOLLAND, 1975) e Busca Tabu (GLOVER e LAGUNA, 1997). Segundo SOUZA e MOCCELLIN (2000), tais métodos consistem de procedimentos de busca no espaço de soluções, definidos por estratégias que exploram apropriadamente a topologia de tal espaço. Segundo este mesmo autor, o sucesso das metaheurísticas se deve a fatores como: i) alusão a mecanismos de otimização da natureza (nos casos do Algoritmo Genético e *Simulated Annealing*); ii) aplicabilidade geral da abordagem, i.e. existe invariavelmente uma estratégia geral de solução, cabendo apenas adaptá-la ao caso específico; iii) facilidade de implementação; e iv) qualidade da solução obtida aliada a um esforço computacional relativamente baixo. YOUSSEF, et al. (2001) também destacaram que o crescente interesse por esta classe de algoritmo pode ser atribuído a suas generalidades, facilidade de implementação e, principalmente, devido à sua robustez na solução de uma grande variedade de problemas de diversas áreas. Estes autores acrescentaram às heurísticas AG, BT e SA as seguintes características: i) são algoritmos aproximativos (heurísticos), i.e., não há garantia de encontrar uma solução ótima; ii) são míopes, i.e. não reconhecem quando a solução ótima é alcançada, o que requer a definição de critérios de paradas estabelecidos pelo projetista e iii) ocasionalmente, aceitam movimentos de pior qualidade.

Segundo REBELLO e HAMACHER (2000), algumas metaheurísticas como os algoritmos genéticos, *simulated annealing*, colônia de formigas etc, são derivadas da natureza e estão numa área limite entre a Pesquisa Operacional e a Inteligência Artificial, com aplicações em vários problemas de otimização. Estes algoritmos foram inspirados na Física, Biologia, Ciências Sociais e operam com um mecanismo de competição e/ou comparação

3. ALGORITMOS GENÉTICOS

Os AGs foram desenvolvidos por John Holland e seu grupo de pesquisa, mas a sua teoria só foi formalizada em 1975 com o aparecimento do livro *Adaptation in Natural and artificial systems* de John Holland (HOLLAND, 1975). Posteriormente, GOLDBERG (1989) aperfeiçoou as idéias desenvolvidas por Holland. Os AGs são métodos de busca e otimização baseados nos princípios de seleção natural e reprodução genética e empregam um processo adaptativo e paralelo de busca de soluções em problemas complexos, se enquadrando na classe de métodos heurísticos inteligentes ou metaheurísticas (REBELLO e HAMACHER, 2000).

Para REBELLO e HAMACHER (2000), os AGs são métodos robustos, que podem ser utilizados para resolver problemas em pesquisa numérica, otimização de funções e aprendizagem de máquina, entre outras aplicações. Esses algoritmos têm sido usados na solução dos mais variados problemas de diferentes domínios, como programação automática, ecologia, Pesquisa Operacional etc (GARCIA et al., 2000). Os AGs têm sido apontados como uma técnica promissora para resolver problemas combinatoriais, como os clássicos problemas de Programação Inteira. Apesar de não garantirem otimalidade, sua grande vantagem está no melhor desempenho ou performance computacional, além de ser um procedimento de relativa simplicidade. Segundo GARCIA et al. (2000), praticamente todos os problemas NP-completos ou NP-árduos têm uma versão de solução heurística que usa esse tipo de algoritmo.

3.1. Princípios básicos

O algoritmo genético inicia o processo de otimização a partir de um conjunto de configurações (população inicial) que pode ser obtida aleatoriamente ou usando algoritmos heurísticos construtivos simples e rápidos. Em cada iteração é obtido um novo conjunto de configurações (nova população) a partir da população corrente usando os operadores de seleção, *crossover* e mutação. Em cada nova iteração são encontradas configurações de melhor qualidade e, eventualmente, nesse processo iterativo pode ser encontrada a solução (configuração) ótima global. De acordo com ROMERO e GALLEGOS (2000), um algoritmo genético realiza uma busca usando

um conjunto de soluções (configurações) e através de um processo iterativo são encontradas novas configurações candidatas. O número de configurações visitadas nesse processo de busca deve ser um número muito reduzido de configurações do espaço de configurações e deve existir uma estratégia adequada para visitar as configurações mais atrativas

Os princípios básicos de um AG, segundo GOLDBARG e LUNA (2000), podem ser resumidos através do seguinte pseudo código:

Algoritmo Genético

- Início

Gerar uma população inicial

Avaliar a *fitness* dos indivíduos da população

Repetir

Início

Selecionar um conjunto de pais na população

Cruzar os pais de modo que se reproduzam

Avaliar a *fitness* dos filhos gerados

Substituir os filhos julgados inadequados

Fim

Até quando critério de parada seja atendido

- Fim

3.2. Elementos do algoritmo genético

Independente da sofisticação de um AG, existem cinco componentes que são básicos em suas implementações (IGNACIO et al. (2000):

- representação das soluções do problema em termos de cromossomas;
- modo de criar a configuração da população inicial;
- uma função de avaliação (aptidão) que permita ordenar os cromossomas, de acordo com a função objetivo;
- operadores genéticos que permitam alterar a composição dos novos cromossomas gerados pelos pais, durante a reprodução; e
- valores dos parâmetros que o AG usa (tamanho da população; probabilidades associadas com a aplicação dos operadores genéticos: critério de seleção, critério de sobrevivência dos cromossomos; taxa de mutação; critérios de parada etc).

A escolha de muitos destes componentes é um processo empírico e depende da experiência e do sentimento do implementador (REBELLO e HAMACHER, 2000).

3.2.1. Codificação das soluções do problema em termos de cromossomas

Na terminologia dos AG's um cromossoma representa um indivíduo na população (i.e. uma configuração ou solução), definido normalmente como um vetor de componentes (GOLBARG e LUNA, 2000). Cada cromossomo apresenta os seguintes elementos: *genes*, *alelo*, *locus* e *fitness* (Figura 3). Os Genes (G) representam as componentes do cromossomo (i.e. uma variável do problema). Os Alelos descrevem os possíveis estados de um atributo do indivíduo (i.e. os possíveis valores de uma variável do problema). Em problemas com variáveis binárias tem-se $G \in (0,1)$. Finalmente, os locus representam a N-ésima posição do atributo no cromossomo (i.e. a posição da componente no vetor de componentes). A *fitness* é a medida de aptidão do indivíduo (cromossomo). Normalmente associada ao valor da função objetivo para uma dada solução.



Figura 3 - Exemplo da estrutura de um cromossomo.

A representação desses cromossomas é apontada pelos especialistas como um fator determinante no sucesso do AG e a forma de codificação mais difundida é a codificação binária. Nesta codificação, cada cromossomo é representado por uma seqüência inteira de bits (Figura 4).

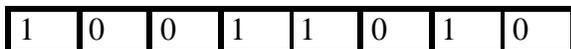


Figura 4 - Esquema de codificação binária de um cromossomo.

Segundo MITCHELL (1996), a maior difusão dessa codificação pode ser atribuída a fatores históricos. Os trabalhos originais de Holland e seus colaboradores

concentravam-se nessa forma de codificação e as aplicações de AG tenderam a seguir essa idéia, uma vez que muitos fundamentos do método como: *crossover*, mutação e outros parâmetros do algoritmo são baseadas nessa codificação. Segundo o mesmo autor, apesar dos fundamentos básicos do AG poderem ser estendidos para outras formas de codificações, nem sempre apresentam boas performances e às vezes não é uma tarefa muito fácil.

Entretanto, em muitas situações a codificação binária é inadequada para representar a solução do problema, como em representação de pesos em redes neurais, seqüências de proteínas, rotas de veículos etc. Portanto, outras formas de codificação devem ser utilizadas, para tais propósitos, como a codificação alfabética e a codificação real. Na codificação alfabética, os alelos são representados por uma seqüência de caracteres alfabéticos, enquanto na codificação real, tais alelos são representados por seqüências de números reais (MITCHELL, 1996) (Figura 5).

Codificação alfabética

A	C	A	T	U	C	A	U
---	---	---	---	---	---	---	---

Codificação real

4	8	0.5	7	1	3	4	8
---	---	-----	---	---	---	---	---

Figura 5- Esquema da codificação real e alfabética de um cromossoma.

A melhor codificação depende muito mais do problema e dos detalhes utilizados no AG, e não há nenhuma forma eficiente de prever qual codificação é mais adequada. Portanto, a experimentação é a melhor forma para comparar o mecanismo de codificação mais adequada para um dado problema (MITCHELL, 1996).

3.2.2. População inicial

O algoritmo genético inicia o processo de otimização a partir de um conjunto de configurações (população inicial). A População em AG refere-se a um conjunto de indivíduos ou um conjunto de soluções do problema (GODBARG e LUNA, 2000). Em relação à população existem dois aspectos que devem ser especificados. O tamanho da população “*n*” e a forma em que é determinada a população inicial. O

tamanho da população deve ser escolhida levando em conta o tamanho do problema e os outros parâmetros genéticos, como a taxa de *crossover* e a taxa de mutação (ROMERO e GALLEGO, 2000).

A forma de gerar uma população inicial pode ser feita de duas maneiras diferentes: de forma aleatória ou utilizando heurísticas (IGNACIO, 2000). Ao se utilizar uma heurística com população aleatória, pode-se aprender muito sobre as características do problema e sobretudo, da evolução deste tipo de população pode-se obter uma configuração satisfatória para o problema de otimização (IGNACIO, 2000). Para algumas aplicações, pode ser conveniente iniciar a população usando-se métodos mais diretos, como os algoritmos gulosos, ou pode-se iniciar com a perturbação de uma solução gerada aleatoriamente.

Durante o processo evolutivo, esta população é avaliada e a próxima geração será uma evolução da anterior, para que isto ocorra, os mais aptos deverão possuir maior probabilidade de serem selecionados para dar origem à nova geração, que deverá ser melhor que a que lhe deu origem. Uma geração mais apta ao ambiente significa uma melhoria no valor da função objetivo. Em cada “passo evolutivo”, conhecido como geração, decodifica-se um indivíduo e avalia-se sua capacidade “reprodutiva”. A partir de então, cria-se uma nova população, por processos de seleção e posterior “cruzamento” de cromossomos da população anterior (GARCIA et al., 2000).

3.2.3. Seleção

Após decidir sobre a codificação a ser adotada e a geração da população inicial ter sido efetuada para o algoritmo genético, a decisão seguinte refere-se a forma de executar a seleção, isto é, como escolher os indivíduos "pais" na população que gerará os novos descendentes ou "filhos" para próxima geração. Uma função de aptidão (*fitness*) é utilizada para quantificar a qualidade genética dos cromossomos pais, e geralmente corresponde à função de custo em problemas de otimização combinatória (IGNACIO, et. al, 2000). De acordo com este autor, esta função de aptidão também é utilizada para decidir se um cromossomo gerado, através de um *crossover*, substitui ou não um cromossomo reprodutor. O propósito de escolher

cromossomos-pais para o cruzamento é incrementar a probabilidade de reproduzir elementos da população que tenha bons valores na função objetivo. Um indivíduo com um valor de aptidão alto possui uma maior probabilidade de contribuir com um ou mais filhos na próxima geração.

Uma seleção muito forte reduz a diversidade da população necessária para sua evolução, ocorrendo a chamada "convergência prematura" do algoritmo, por outro lado, seleção muito fraca resultará em evolução muito lenta da população, o que significa uma convergência lenta do algoritmo (MITCHELL, 1996). De acordo com GARCIA et al (2000), no processo de otimização quando se implementa o algoritmo genético, a seleção é a etapa mais crítica, pois é a que consome mais tempo com cálculos da função de aptidão e probabilidade de sobrevivência de cada indivíduo. Desta forma, quanto maior o número de indivíduos, maior tempo será gasto no processo de seleção, e conseqüentemente, a velocidade de convergência do algoritmo genético está relacionada diretamente como o tamanho da população. Ainda, segundo os mesmos autores, grande parte do custo computacional do algoritmo está concentrado na fase de seleção.

Diversos esquemas de seleção tem sido propostos na literatura de AG. A seguir são descritos alguns dos métodos mais comuns. Tais descrições não provêm diretrizes rigorosas para prevê a adequacidade e eficiência dos métodos para um determinado problema. Esta é ainda uma questão aberta para os AGs, sendo que a experiência e os ensaios são ainda as melhores formas para selecionar um dado método. Existem vários métodos de seleção disponíveis para os AGs. Alguns desses métodos são: a seleção proporcional, seleção por escalonamento, seleção de *Boltzmann*, seleção por ranqueamento, seleção *tournament* e o elitismo, conforme descritos por MITCHELL (1996).

Na seleção proporcional, cada indivíduo i possui uma expectativa de sobrevivência (E_i) associada ao seu valor de aptidão, geralmente dada por:

$$E_i = \frac{f_i}{\sum_{i=1}^n f_i} \quad (1)$$

em que:

f_i = função objetivo avaliada no indivíduo i ; e

n = tamanho da população.

Ao se utilizar a seleção proporcional, deve-se evitar a existência de "super indivíduos" no início do processo, proporcionados por cromossoma que possuem probabilidades de sobrevivência muito maiores do que a média da população (MITCHELL, 1996 e GARCIA et al., 2000). A existência desses indivíduos é indesejável, uma vez eles podem dominar o processo de seleção e reprodução e nesse caso, a população inteira pode convergir para uma solução logo no início do processo. Essa situação recebe o nome de "Convergência Prematura" (MITCHELL, 1996 e GARCIA et al., 2000).

A seleção por escalonamento utiliza a função *sigma scaling* para calcular a expectativa de sobrevivência dos indivíduos, dada por:

$$E_{it} = \begin{cases} 1 + \frac{f(i) - \bar{f}(t)}{2s(t)} & , \text{ se } s(t) \neq 0 \\ 1 & \text{ se } s(t) = 0 \end{cases} \quad (2)$$

em que:

E_{it} = expectativa de sobrevivência do indivíduo i no tempo t ;

$f(i)$ = *fitness* do indivíduo i ;

$\bar{f}(t)$ = *fitness* média da população no tempo t ; e

$s(t)$ = desvio padrão da *fitness* da população no tempo t .

Segundo MITCHELL (1996), a vantagem desse método é evitar a convergência prematura mantendo a pressão de seleção constante durante toda a busca. No início da corrida, onde o desvio padrão da população é geralmente maior, o método penaliza os indivíduos que apresentam *fitness* muito acima da média, reduzindo dessa forma as diferenças de probabilidade de sobrevivência entre indivíduos mais e menos aptos, propiciando dessa forma maiores chances de reprodução para os indivíduos menos aptos, o que aumenta a diversidade da população. Igualmente, no fim da corrida, onde geralmente reduz-se o desvio padrão

da população, os indivíduos mais aptos terão maior chance de sobrevivência, o que permite a convergência do algoritmo.

A seleção por escalonamento mantém a pressão de seleção constante ao longo da corrida. Entretanto, segundo MITCHELL (1996), freqüentemente variação na pressão de seleção ao longo da corrida pode ser uma importante estratégia para melhorar a performance do AG. Por exemplo, no início da corrida poder-se-ia manter uma baixa pressão de seleção, permitindo a reprodução de indivíduos menos aptos, caracterizando uma fase de maior diversificação da população. No final da corrida, aumentaria a pressão de seleção, selecionando indivíduos mais aptos para reprodução, caracterizando um processo de intensificação da busca em regiões mais promissoras.

Uma forma de implementar esta estratégia é utilizar a "Seleção de Boltzmann", segundo MITCHELL (1996), uma abordagem similar a *simulated annealing*, na qual a "temperatura" controla continuamente a pressão de seleção. A temperatura inicia alta, o que resulta em uma baixa pressão de seleção (i.e., cada indivíduo tem uma probabilidade razoável de reproduzir). A temperatura é diminuída gradualmente, de acordo com um esquema prefixado, o que gradualmente aumenta a pressão de seleção, assim permitindo o AG intensificar a busca em regiões mais promissoras, propiciada pela maior probabilidade de sobrevivência de indivíduos mais aptos.

A seleção de Boltzmann pode ser implementada da seguinte forma (MITCHELL, 1996):

$$E_{it} = \frac{e^{f(i)/T}}{E_t} \quad (3)$$

em que:

T = a temperatura;

$$E_t = \sum_{i=1}^n e^{f(i)/T} / n \quad (4)$$

E_t = média da população no tempo t ;

n = tamanho da população, e demais parâmetros conforme já definidos.

A seleção por ranqueamento ou ordenamento é um outro método de seleção alternativo, cujo propósito é também prevenir convergência rápida do AG (MITCHELL, 2000). Segundo este mesmo autor, na versão proposta por Baker em 1985, os indivíduos na população são ranqueados de acordo com o seu *fitness*, e a probabilidade de sobrevivência de cada indivíduo depende de seu *rank* ao invés de seu *fitness* absoluto. As vantagens deste método, segundo MITCHELL (1996) é que ele diminui a pressão de seleção permitindo que indivíduos mais aptos dominem o processo de reprodução, uma vez que os valores absolutos das *fitness* não são considerados, e sim a sua posição no rank. As desvantagens são que, com a baixa pressão de seleção, pode haver problemas de convergência do algoritmo e o maior custo computacional necessário para classificar (ranquear) os indivíduos em cada geração.

Segundo MITCHELL (1996), o método de ranqueamento linear proposto por Baker é o seguinte: cada indivíduo na população é ranqueado em ordem crescente de *fitness*, de 1 a n . O usuário escolhe um valor esperado Max do indivíduo com rank n , com $Max \geq 0$. A expectativa de sobrevivência de cada indivíduo i na população para o tempo t é então dado por:

$$E_{it} = Min + (Max - Min) \frac{rank_{it} - 1}{n - 1} \quad (4)$$

em que:

Min = valor esperado do indivíduo com rank igual a 1.

Segundo MITCHELL (1996), dado a restrição de que $Max \geq 0$ e $\sum_i E(i,t) = n$ (uma vez que o tamanho da população mantém constante de geração a geração), pode-se mostrar que: $1 \leq Max \leq 2$ e $Min = 2 - Max$. Ainda de acordo com MITCHELL (1996), Baker recomenda $Max = 1.1$.

Na seleção *tournament*, também conhecida como seleção por jogos, dois indivíduos são escolhidos ao acaso na população. Escolhe-se um parâmetro k (por exemplo 0.75) e, em seguida, um número aleatório r é então escolhido entre 0 e 1. Se $r < k$, os dois indivíduos são selecionados para serem pais; caso contrário, eles são recolocados na população original e podem ser selecionados novamente. As vantagens deste método, além de sua simplicidade, é a economia no processamento

de operações como cálculo de funções *fitness*, cálculo de médias, desvio padrão, classificação de indivíduos etc, típicos dos outros métodos de seleção. Com isso, um menor esforço computacional é requerido. Entretanto, por requerer poucas informações do sistema pode aprender pouco sobre as suas características, resultando em baixa performance do algoritmo em termos da qualidade da solução obtida (MITCHELL, 1996).

A seleção por elitismo força o AG a reter alguns dos melhores indivíduos em cada geração (De Jong (1975), citado por MITCHELL, 1996). O objetivo desse método é evitar a perda de bons indivíduos quando submetidos a *crossover* ou mutação.

3.2.4. Implementação da seleção

A operação de reprodução ou seleção é um processo em que copiam-se indivíduos de acordo com seus valores de aptidão, dados por uma função de avaliação ($f(x)$). O operador de seleção pode ser implementado de diversas maneiras, sendo considerada a mais fácil a que é denominada “*roleta russa*”, conforme mostrado por GARCIA et al. (2000).

O método da roleta pode ser implementado da seguinte forma (MITCHELL, 1996): cada indivíduo é assinalado numa roleta com numa fatia proporcional ao seu *fitness*. A roleta é girada n vezes, onde n é o número de indivíduos da população. Em cada giro, o indivíduo sobre a marca da roleta é selecionado para estar no grupo dos pais da próxima geração.

3.2.5. Operadores genéticos

Operadores genéticos são as regras que permitem a manipulação dos cromossomos através do cruzamento (*crossover*) e mutação (GOLDBARG e LUNA, 2000).

O operador de *crossover* permite a obtenção de cromossomos (indivíduos) filhos (*offspring*), a partir da combinação (cruzamento) dos cromossomos pais. O procedimento consiste em escolher dois cromossomos para serem pais de dois novos

cromossomos filhos, os quais terão maior probabilidade de ter uma melhor imagem quando aplicados a função objetivo. O operador de *crossover*, juntamente com a mutação, é responsável pela evolução dos indivíduos. O cruzamento é feito na esperança de que dois pais considerados aptos transmitam a seus filhos as suas melhores características, gerando indivíduos melhores do que eles; i.e., a combinação das melhores características dos cromossomo pais possuem maiores chances de produzir uma melhor configuração (solução) para o problema (REBELLO e HAMACHER, 2000).

Existe basicamente três formas de executar a operação de *crossover*: em um único ponto, em dois pontos, em múltiplos pontos (MITCHELL, 1996).

No *crossover* de um ponto, um único ponto é escolhido ao acaso e o material genético dos dois cromossomos pais são trocados a partir desse ponto para criar os dois filhos. No *crossover* de dois pontos, escolhe-se aleatoriamente dois pontos de ruptura nos cromossomos pais e o material genético entre estes pontos são trocados para gerar dois novos cromossomos filhos. Finalmente, no *crossover* de múltiplos pontos (ou multipontos) vários pontos são escolhidos aleatoriamente e as trocas dos materiais genéticos dos cromossomos pais entre esses pontos são efetuadas de acordo com uma regra preestabelecida.

Existe um quarto tipo de *crossover*, conhecido com *crossover* uniforme ou o sobrecruzamento (IGNACIO et al., 2000). Segundo estes autores, neste tipo de *crossover*, um pai é escolhido aleatoriamente para que dê origem, com seu primeiro bit, ao primeiro filho, o segundo filho terá origem em seu segundo bit e assim sucessivamente. Este processo continua até que todos os bits tenham sido alocados.

A operação de *crossover* é validada da seguinte forma: uma população composta por n indivíduos é agrupada aleatoriamente em pares, a fim de gerar um conjunto de $n/2$ progenitores (“pais”) potenciais, que são escolhidos para os cruzamentos. Atribui-se a cada par de pais uma probabilidade de cruzamento (P_C). Gera-se para cada par, um número aleatório entre 0 e 1. Em seguida, compara-se o valor gerado com a probabilidade P_C . Se o número encontrado for inferior a P_C , o cruzamento é permitido, caso contrário, os progenitores são mantidos inalterados e passados para a próxima geração (GARCIA et al, 2000).

O operador genético mutação permite a produção de um novo indivíduo por alterações aleatórias diretas no cromossomo. Segundo IGNACIO et al (2000), ainda que por muito tempo a mutação tenha sido concebida com um operador secundário, ela representa um aspecto importante do AG. De acordo com estes autores, o objetivo da operação de mutação é manter a diversidade na população, evitando convergência rápida para ótimos locais (convergência prematura). A mutação confere também aos AGs capacidade de recuperação de informações que poderão ser valiosas. Ainda, segundo GARCIA et al. (2000), o operador de mutação deve ser usado com cautela, pois uma taxa de mutação alta aumenta a possibilidade de um indivíduo com boas características de ser destruído.

A mutação é uma alteração aleatória do valor de uma posição do cromossomo, ou seja, o valor de um determinado gene do cromossomo é invertido. No caso binário, a mutação consiste em substituir, com probabilidade P_m (taxa de mutação), o valor de um bit. Para outros tipos de codificação geralmente é possível definir outras alternativas de mutação (ROMERO e GALLEGO, 2000). A mutação não passa por testes de aptidão uma vez que o objetivo é introduzir diversidade à população (IGNACIO, 2000).

3.2.6. Parâmetros de controle de um algoritmo genético

A eficiência de um algoritmo genético é altamente dependente de seus parâmetros de controles, tais como o tamanho da população, taxa de *crossover* e a taxa de mutação. Existem muitos trabalhos desenvolvidos com a finalidade de dimensionar os valores dos parâmetros na literatura de computação evolucionária (MITCHELL, 1996). Entretanto, não há resultados conclusivos sobre os melhores valores desses parâmetros para todas as aplicações. Muitas aplicações usam valores reportados em outros trabalhos, outras definem os seus próprios valores com base em experimentação. Segundo GARCIA et al. (2000), há consenso na literatura de AG que, em geral, o tamanho de cada população não deve ser menor que 25, a probabilidade de cruzamento deve estar compreendida entre 0,5 e 0,95 e probabilidade de mutação ser menor que 0,01. Segundo MITCHELL (1996), as

abordagens que parecem bastantes promissoras é ter os valores dos parâmetros *adaptados* em tempo real (continuamente) na busca. Segundo esse mesmo autor, diversas abordagens adaptativas têm sido focos das nas pesquisas de computação evolucionária.

3.2.7. Formalização do algoritmo genético

Segundo GOLDBARG e GOUVÊA (2000), um AG operacionaliza seu processo evolucionário através do controle de adequação dos indivíduos da população (cromossomos), sendo a avaliação de adequação um dos elementos centrais para a promoção da adaptação. Estes autores definiram formalmente um AG da seguinte forma:

$$AG = (N, P, F, \Theta, \Omega, \Psi, \tau)$$

Onde P é uma população de N indivíduos, $P = (S_1, S_2, \dots, S_N)$. Cada indivíduo S_i , $i = 1, \dots, N$ é uma cadeia ou um conjunto de valores inteiros de comprimento n representando uma solução do problema. F representa a função *fitness* ou *adequação* que retorna um valor positivo e real na avaliação de cada indivíduo.

$$F: S_i \in \mathbb{A}^n, i = 1, \dots, N$$

Θ é o operador de seleção de pais que escolhe r indivíduos de P

$$\Theta: P \in \{p_1, p_2, \dots, p_r\}$$

Ω é um conjunto de operadores genéticos que inclui o operador de *crossover* (denominado Ω_C), o operador de *mutação* (denominado Ω_M) e qualquer outro operador específico que produzem s filhos a partir de r pais.

$$\Omega = \{ \Omega_C, \Omega_M, \dots \} : \{p_1, p_2, \dots, p_r\} \rightarrow \{f_1, f_2, \dots, f_s\}$$

Ψ é o operador de remoção que retira s indivíduos selecionados na população P , permitindo, por exemplo, que s filhos (novos indivíduos) sejam adicionados à nova população P_{t+1} :

$$P_{t+1} = P_t - \Psi(P_t) + \{f_1, f_2, \dots, f_s\}$$

τ é o critério de parada:

$$\tau: P_t \in \{\text{verdadeiro}, \text{falso}\}$$

4. BUSCA TABU

A Busca Tabu (BT) teve início em procedimentos combinatórios em problemas de cobertura não lineares no fim da década de 70. As idéias básicas de sua forma atual foram desenvolvidas por GLOVER (1977). Tentativas adicionais de formalização foram relatadas por GLOVER (1989), WERRA e HERTZ (1989) e GLOVER (1997).

A BT é aplicada a diversos tipos de problemas combinatoriais. Os primeiros foram: sequenciamento da produção, balanceamento de canais de comunicação e análise de agrupamentos. Recentemente, ela tem sido aplicada ao problema do caixeiro viajante, coloração de grafos, projeto de circuitos integrados, roteamento de veículos, dentre outros (GLAZAR, 2000).

4.1. Filosofia do método

Busca tabu é uma metaheurística que guia um procedimento de heurística local, com a finalidade de explorar o espaço de solução além da otimalidade local. O procedimento local é uma busca que utiliza uma operação chamada movimento para definir uma vizinhança de qualquer solução. A busca tabu reduz a vizinhança ao classificar certos movimentos como proibidos ou tabu (YAMASHITA et al., 2000). Este mecanismo previne ciclagem, i.e., a execução repetitiva da mesma seqüência de movimentos, e direciona a busca para regiões ainda não exploradas. A imposição de restrições sobre atributos dos movimentos ou transições é a base para superação da otimalidade local. A característica relevante da Busca Tabu refere-se à capacidade de não ficar restrita às soluções ótimas locais, explorando os vales do espaço de soluções na tentativa de obtenção da solução ótima global (GLOVER e LAGUNA, 1997).

Um outro elemento fundamental na BT é o uso flexível de memória, que engloba o processo de criar as estruturas que permitam explorar eficientemente o histórico de cada solução gerada, fazendo com que as informações obtidas anteriormente sirvam como parâmetros para a análise e

geração de novas soluções. O uso da memória flexível permite guiar o processo de busca no conjunto de soluções do problema, fazendo repetidos movimentos de uma solução para outra solução vizinha, determinando alternativas de busca a serem qualificadas como proibidas (tabu) ou atrativas (GLOVER e LAGUNA, 1997).

4.2. Princípios básicos

Uma descrição completa dessa técnica pode ser encontrada em GLOVER e LAGUNA (1997). Para estes autores, a BT pode ser aplicada diretamente à declarações verbais ou simbólicas de muitos tipos de problema de decisão, sem a necessidade de transformá-lo em formulações matemáticas. Contudo, a notação matemática é mais útil para representar os princípios básicos da técnica. GLAZAR (2000), descreve a BT, definindo-se um problema de otimização combinatoria da seguinte forma:

$$\begin{array}{ll} \text{Minimize } Z = f(x) & \\ \text{s.a. } & x \in X \end{array} \quad (5)$$

No problema de otimização acima, a função objetivo de minimização (ou maximização) pode ser linear ou não linear. O conjunto X representa o vetor de restrições das variáveis de decisão x . As restrições podem incluir desigualdades linear ou não linear, e x pode assumir valores contínuos ou discretos.

Seguindo o modelo acima, BT explora iterativamente o conjunto de soluções, denotado por X , através de repetidos movimentos de uma solução x' localizada na vizinhança $N(x)$ de x . Com este procedimento, a BT inicia da mesma forma que um método de busca local ou busca em vizinhança, procedendo iterativamente de um ponto (solução) para outro até que um critério de parada escolhido seja satisfeito. Cada $x \in X$ tem uma vizinhança $N(x) \subseteq X$, e cada solução $x' \in N(x)$ é alcançada a partir de x por uma operação chamada de *movimento* (GLOVER e LAGUNA, 1997).

A base da BT é uma busca local, que resolve problemas de otimização local, encontrando uma solução x' que seja o $\min \{f(x') / x' \in V^* \subseteq N(x)\}$, para problemas

de minimização, onde V^* é um subconjunto da vizinhança de x (Glover, 1989; citado por GLAZAR, 2000).

Contrastando a BT com um método descendente simples, onde a meta é minimizar $f(x)$ (ou a um método ascendente correspondente a maximizar $f(x)$), o método permitiria somente mover para a solução vizinha que melhorasse o valor da função objetivo corrente e terminaria quando nenhuma solução melhorada pudesse ser encontrada. Um pseudo código de um método descendente genérico foi apresentado por (GLOVER e LAGUNA, 1997):

- escolher $x \in X$ para iniciar o processo;
- encontre $x' \in N(x)$ tal que $f(x') < f(x)$;
- se nenhum x' melhor pode ser encontrado, x é o ótimo local e o método para;
- caso contrário, escolha x' para ser o novo x e volte ao passo 2.

O x final obtido pelo método descendente é chamado de ótimo local, desde que ele seja melhor que todas as soluções na sua vizinhança. Uma deficiência evidente do método descendente é que o ótimo local em muitos casos não será um ótimo global. i.e., ele usualmente não minimiza $f(x)$ para todo $x \in X$. A capacidade em escolher soluções boas a partir da vizinhança corrente é ampliada pelos mecanismos de guia da BT introduzidos para ir além do ponto de terminação do ótimo local. A seguir, é apresentado o algoritmo de BT básico, onde $f(x)$ é a função objetivo e T é a Lista Tabu (IGNACIO et al., 2000):

Início

$$k=0$$

$$f(x^*)=\text{infinito}$$

$$T=\emptyset$$

Obter uma solução inicial $x^k \in X$

Faça enquanto as condições de término não forem satisfeitas

$$k=k+1$$

Gere o movimento $y=m \otimes x$

Calcule o valor do movimento

Escolha o movimento não tabu ou que atenda aos critérios de aspiração

Atualize as condições de término.

Se as condições de término não foram satisfeitas

Execute o movimento escolhido, $x^k = y$

Calcule $f(x^k)$

Se $f(x^k) < f(x^*)$

$$x^* = x^k$$

Atualize T

Fim

A compreensão desses mecanismos passa a ser de crucial importância para o sucesso da implementação da BT. Entretanto, não se pretende esgotar aqui todas as discussões a cerca do assunto, tampouco listar minuciosamente todos os mecanismos disponíveis para as implementações de BT. O objetivo dos tópicos seguintes é dar uma visão geral dos mecanismos básicos descritos na literatura especializada. Maiores detalhes podem ser encontrados nos trabalhos originais de GLOVER e LAGUNA (1997) e na ampla revisão desses mecanismos feita por GLAZAR (2000). Além desses, atualmente vários outros trabalhos tratam do assunto em aplicações variadas.

4.3. Vizinhanças e movimentos

Conforme GLAZAR (2000), seguindo os princípios básicos da BT, define-se um movimento m como sendo uma função que transforma uma solução x em outra solução x' ($x' = x \oplus m$: esta notação representa a mudança da solução x para a solução x' através do movimento m), onde $x \in N(x)$ e $N(x) \subset X$ é a vizinhança de x . Logo, uma vizinhança $N(x)$ de x pode ser obtida por:

$$N(x) = \{x' \mid \exists m \in M(x), \text{ com } x' = x \oplus m\} \quad (6)$$

em que $M(x)$ é o conjunto de m movimentos legais para transformar x em x' .

A função primordial do movimento m é gerar a uma vizinhança $N(x,k)$ a partir de uma solução x_k , onde x é uma solução para o problema e k é a iteração corrente. Os mecanismos de movimentos são utilizados para fugir de ótimo local e devem ser habéis para aceitar um movimento que leve de x para $x' \in N(x,k)$ mesmo quando $f(x') > f(x)$, em caso de problemas de minimização. Em contrapartida, se aceitarmos movimentos desta natureza, sem tomarmos algumas precauções, podem ocorrer ciclos, onde o processo retorna repetidamente a uma solução já visitada. A BT lida com este problema incorporando um estrutura de memória que proíbe ou penaliza certos movimentos que poderiam retornar a uma

solução recentemente visitada. Tais movimentos proibidos são chamados de *movimento tabu*. Proibindo ou penalizando certos movimentos, a pesquisa fica restringida a um subconjunto de soluções $V^* \subset N(x)$. Com isso, cada solução é visitada uma única vez, evitando, portanto, ciclos. Entretanto, nem sempre que se retornar a uma solução visitada, não necessariamente forma-se um ciclo. É importante, em certos instantes, a busca retornar a soluções já visitadas e continuar em uma direção, talvez mais promissora. A BT emprega movimentos que são reversíveis, ou seja

$$\forall m \in M(x), \exists m^{-1} \in M(M(x)) \text{ tal que } (x \oplus m) \oplus m^{-1} = x. \quad (7)$$

Assim, os movimentos tabu são proibidos dentro de um intervalo k de interações definido como *permanência tabu* ($|T|$) do movimento. Isto significa que uma solução recentemente visitada é proibida de ser revisitada por um período $|T|$ de iterações, prevenindo ciclos menores ou iguais a $|T|$ (GLAZAR, 2000).

Vários são os mecanismos de movimentos utilizados para gerar uma vizinhança. Eles podem ser utilizados individualmente ou de forma mesclada, variando a cada iteração (ou intervalo de iterações), aleatoria ou deterministicamente. A variação destes movimentos em uma determinada busca, se bem aplicada, pode alternar entre uma estratégia de diversificação e intensificação (GLAZAR, 2000).

4.4. Memória e classificação tabu

Existem dois tipos de memória na BT: memória de curto prazo e memória de longo prazo. Cada tipo de memória é acompanhado pelas suas estratégias especiais. Entretanto, o efeito de ambos os tipos de memória pode ser visto com modificadores da vizinhança $N(x)$ da solução corrente x .

A BT utiliza um conjunto de princípios de memória flexível e exploração sensível (IGNACIO et al., 2000). Segundo os mesmos autores, dentro da memória flexível, podem-se considerar os seguintes aspectos: (1) *Seletividade* - estratégias de esquecimento das informações que não são relevantes para a busca. (2) *Abstração e Decomposição* - obtém-se guardando as informações explicitamente e por atributos.

(3) *Tempo* - consideram-se eventos recentes, eventos freqüentes e se diferenciam conceitos de curto e longo prazos. (4) *Qualidade e impacto* - atração relativa das escolhas alternativas e a magnitude de trocas. (5) *Contexto* - considera-se para a análise a dependência regional, estrutural e seqüencial das vizinhanças. Dentro da exploração sensível, ainda de acordo com IGNACIO et al. (2000) podem-se considerar: *condições tabu e níveis de aspiração* - imposições estratégicas de limitações e induções; *intensificação* – enfoque concentrado em boas regiões e boas características das soluções; *diversificação* - caracteriza e pesquisa novas regiões promissoras; *oscilação estratégica* - padrões de busca que aproxima e transpassa um nível crítico repetidamente em direções diferentes; e *re-encaminhamento e extensão de soluções* - integração e extensão das soluções.

O uso da memória flexível emprega a criação e exploração da estrutura para tirar vantagens do histórico da busca. Os itens que geralmente são armazenados são as configurações das soluções, o correspondente valor da função objetivo e o número da interação na qual as soluções foram encontradas. As estruturas de memória mais utilizadas são a lista FIFO, *hash e árvore digital* (GLAZAR, 2000).

4.4.1. Memória de curto prazo

Devido à exploração de memória, $N(x,k)$ depende da trajetória seguida pelo movimento de uma solução para a próxima. Como ponto de partida para examinar memória de curto prazo, considere a forma de memória incluída numa *lista tabu* T que explicitamente contém t diferentes soluções, isto é, $T = \{x_1, x_2, \dots, x_t\}$, assim pode-se obter $N(x, k) = N(x) - T$, onde $N(x,k)$ consiste de soluções de $N(x)$ não incluída na lista tabu. A memória de curto prazo mais utilizada guarda os atributos das soluções que tem mudado durante o passado mais recente, e é chamada de *memória baseada na recência*. Este é o tipo de memória mais comum na maioria das descrições de BT na literatura. Nessa estrutura de memória, atributos selecionados que ocorreram em soluções recentemente visitadas são classificados como *tabu-active* durante um período de k iterações, conforme já definido, como sendo a permanência tabu ($|T|$) do movimento. Soluções que contém elementos *tabu-active*, são aquelas que tornaram-se tabu (GLOVER e LAGUNA, 1997). Isto

previne certas soluções visitadas no passado recente de pertencerem a $N(x,k)$ e consequentemente, de serem revisitadas durante um período $|T|$. Aos movimentos que conduzem a soluções tabu denomina-se *movimento tabu*.

Independente dos detalhes de implementação, funções de memória de curto prazo constituem-se um dos importantes fundamentos da BT. Tais funções dão à busca a oportunidade para continuar além do ótimo local, por permitir a execução de movimentos não melhorados juntamente com a modificação da estrutura de vizinhança de soluções subsequentes (GLAZAR, 2000).

4.4.1.1. Permanência tabu

O intervalo k de iterações que um movimento permanece tabu é chamado de *permanência tabu* ou simplesmente *permanência*. A escolha do valor de permanência mais apropriado pode ter grande efeito na performance da Busca Tabu. Valores muito pequenos para k podem permitir que ocorra ciclagem no processo de busca e estacionamento em ótimos locais, enquanto que valores elevados, apesar de direcionar a busca para novas regiões, pode piorar a qualidade da solução devido à redução de movimentos alternativos (GLOVER e LAGUNA, 1997). De acordo com estes autores, a permanência tabu pode variar com diferentes tipos ou combinações de atributos e pode também variar ao longo de diferentes intervalos de tempo ou estágios da busca. Ainda, segundo estes autores, a permanência tabu tem-se mostrado depender do tamanho da instância do problema. Entretanto, nenhuma regra simples tem sido designada para prover uma permanência efetiva para todas as classes de problema.

Devido à importância deste parâmetro, várias formas de defini-lo e operá-lo têm sido propostas na literatura conforme pode ser encontrado em GLOVER e LAGUNA (1997). Há basicamente dois tipos de permanência: permanência estática e permanência dinâmica. A permanência estática, como próprio nome sugere, o tamanho da lista é fixado e permanece inalterado durante toda a busca. Consequentemente, na permanência dinâmica, o tamanho da lista é alterado ao longo do curso da busca.

De acordo com (GLOVER e LAGUNA, 1997), a permanência dinâmica pode ser implementada por duas formas distintas: aleatoriamente e sistematicamente. Ambas as formas usam um intervalo de permanência definido pelo parâmetro t_{min} e t_{max} . A permanência tabu t é selecionada aleatoriamente no intervalo definido, usualmente, seguindo uma distribuição uniforme. Na permanência aleatória, o valor escolhido em uma dada iteração é mantido constante por αt_{max} iterações, e então uma nova permanência é selecionada pelo mesmo processo. A determinação dos extremos t_{min} e t_{max} da distribuição uniforme são parâmetros importantes para propiciar uma duração tabu balanceada, para evitar ciclagem e degeneração da busca.

A permanência tabu dinâmica baseada no esquema aleatório é atrativo por sua facilidade de implementação. Entretanto, o esquema totalmente aleatório pode não ser a melhor estratégia quando informações específicas sobre o contexto estão disponíveis. GLOVER e LAGUNA (1997) mencionam que em alguns casos pode ser interessante alternar entre valores altos e baixos de t , permitindo a indução de padrões de diversidade. Segundo os mesmos autores, uma forma simples de implementar a permanência dinâmica sistemática consiste na criação de uma sequência de valores de permanência para a busca tabu no intervalo definido por t_{min} e t_{max} . Assim, por exemplo, uma sequência alternada {5, 8, 6, 9, 7, 10} poderia ser usada ao invés da distribuição uniforme, para assinalar o valor corrente da permanência. Desta forma, o procedimento iniciaria com a permanência t_{min} igual a 5 mantendo-a por um intervalo de α iterações; e sequencialmente escolheria novos valores de t até alcançar t_{max} igual a 10. O procedimento repetiria até que a regra de parada do algoritmo fosse acionada.

Sistemas adaptativos bem desenhados têm sido apontados como importantes artifícios para reduzir ou mesmo eliminar a necessidade de intuição na definição do tamanho da permanência. Tais sistema podem ser desenhados, por exemplo, para variar a permanência tabu durante a busca, provendo formas de definir escolhas mais inteligentes desse parâmetro ao longo da busca. Estes estudos constituem em importantes áreas de estudo (GLOVER e LAGUNA, 1997).

4.4.2. Memória de longo prazo

As abordagens de memória de curto prazo sozinhas, muitas vezes, proporcionam a obtenção de soluções de alta qualidade. Conseqüentemente, muitos trabalhos envolvendo a aplicação de BT têm-se restringido suas aplicações apenas com considerações de curto prazo. Entretanto, em geral, a BT torna-se mais robusta com a inclusão de memória de longo prazo em suas estratégias de busca (GLOVER e LAGUNA, 1997). Nas estratégias de memória de longo prazo, a vizinhança modificada produzida pela busca tabu pode conter soluções geralmente consistindo de soluções elite selecionadas (ótimo local de alta qualidade) encontrada em vários pontos do processo de soluções. O uso de memória de longo prazo, além de melhorar a qualidade da solução, pode diminuir a necessidade de longas corridas do algoritmo para encontrar soluções de melhor qualidade, reduzindo-se o esforço computacional. (GLOVER e LAGUNA, 1997).

As principais estratégias de memória de longo prazo são descritas minuciosamente em GLOVER e LAGUNA, 1997 e são resumidas a seguir.

4.4.2.1. Abordagem baseada na frequência

Memória baseada na frequência produz um tipo de informação que complementa a informação produzida pela memória baseada na recência, ampliando a base para selecionar movimentos preferidos. Esta estratégia armazena as informações sobre a frequência das trocas ao longo do processo de busca. Existem duas formas de memória baseada na frequência: *frequência de transição* e *frequência de residência*. A frequência de transição expressa o número de iterações onde um atributo entra ou deixa uma solução visitada ao longo da trajetória da busca, enquanto frequência de residência expressa o número de iterações onde um atributo é membro de uma solução visitada em uma trajetória particular. A frequência de transição indica a frequência em que atributos mudam (entram ou deixam a solução), enquanto frequência de residência indica a frequência com que atributos são membros das soluções geradas. Tais frequências são, muitas vezes, confundidas (ou tratadas identicamente) na literatura.

(GLOVER e LAGUNA, 1997). Segundo estes autores, uma distinção notável é que medidas de residência, em contraste com medidas de transição, não está preocupada com as mudanças dos atributos de uma solução particular ou se eles são atributos que mudam com o movimento de uma solução para outra. Utilizando como exemplo, o problema da árvore k mínima, os mesmos autores salientam que a medida de residência poderia contar o número de vezes que um arco (i,j) faz parte da solução, enquanto a medida de transição poderia contar o número de vezes que um arco (i,j) foi adicionado ou removido da solução. Uma alta frequência de residência pode indicar que um atributo é altamente atrativo se o domínio consiste de soluções de alta qualidade, ou pode indicar o oposto, se o domínio consiste de soluções de baixa qualidade (GLOVER e LAGUNA, 1997).

4.4.2.2. Estratégias de intensificação

As estratégias de intensificação restringem a vizinhança para favorecer soluções com propriedades que ocorreram em boas soluções previamente visitadas, ou em subconjuntos de boas soluções. Concentra-se a pesquisa na vizinhança das melhores soluções visitadas. Esta estratégia resulta em uma diminuição do esforço computacional no processo de busca. Em contrapartida, a busca pode ficar confinada numa região onde se encontra um ótimo local, podendo não atingir o ótimo global (GLOVER e LAGUNA, 1997).

4.4.2.3. Estratégias de diversificação

Métodos de busca baseados em otimização local freqüentemente confiam em estratégia de diversificação para aumentar sua efetividade em explorar o espaço de solução definido pelos problemas de otimização combinatorial. Algumas dessas estratégias são desenhadas com o propósito principal de prevenir o processo de busca da *ciclagem*. Outras estratégias são introduzidas para conceder robustez ou vigor adicionais para a busca. Na BT, a diversificação pode ser criada até certo ponto por funções de memória de curto prazo, como a alternância do valor da permanência ao longo da busca. Mas é particularmente reforçada por certas formas

de memória de longo prazo. Estratégias de diversificação em BT, como seu nome sugere, são desenhadas para direcionar a busca para novas regiões, ou seja, redirecionar a busca para regiões ainda não exploradas. Uma forma simples de implementar estratégias de diversificação consiste em modificar a função objetivo de modo a tornar menos atrativos valores próximos e mais atrativos valores afastados da vizinhança. Com esta estratégia é mais difícil encontrar ótimos locais (GLAZAR, 2000). Segundo GLOVER e LAGUNA (1997) as estratégias de intensificação são baseadas na modificação das regras de escolha dos movimentos e por processo de reinicialização. No primeiro caso, são efetuadas modificações nas regras de escolha para encorajar combinações de movimentos e características de soluções historicamente encontradas boas. No segundo caso, um processo de reinicialização propicia um retorno a regiões atrativas para explorar o espaço de soluções mais profundamente.

Segundo GLOVER e LAGUNA (1997), o uso combinado das estratégias de intensificação e diversificação pode tornar a BT ainda mais eficiente. Num horizonte de longo prazo, esta combinação pode permitir a busca alternar entre uma seqüência de passos que intensificam a busca em uma região promissora e passos que diversificam a busca alcançando regiões com características contrastantes. Isto pode ser obtido intensificando a busca numa região, até que um determinado número de iterações seja executado, sem que haja melhora na melhor solução até então encontrada. A partir daí, diversifica-se a busca para outras regiões.

4.5. Lista tabu

A lista tabu é representada por uma matriz onde são armazenadas o *status* tabu do movimento. Um movimento considerado tabu é proibido de ser gerado por um número de iterações iguais à sua permanência tabu, evitando que a busca retorne a soluções recentemente visitada. Não existe uma regra universal para definição da estrutura da lista tabu e nenhuma estrutura que seja melhor par todas as aplicações. A estrutura dessa lista depende muito mais do problema em questão. Entretanto, algumas regras podem ser levadas em consideração no momento em

que se planeja a estrutura da lista (GLOVER e LAGUNA, 1997). Segundo estes mesmos autores, se o tamanho da vizinhança é pequeno o suficiente para armazenar um item de informação para cada atributo do movimento usado para definir uma restrição tabu, em geral é vantajoso armazenar o número da iteração que identifica quando a restrição tabu será descartada. Isto faz com que o estado tabu de um movimento seja testado em tempo constante.

4.6. Atributos de movimento

Para transformar de uma solução x em outra solução x' através de um mecanismo de movimento, utiliza-se os *atributos de movimento*. Segundo Glover e Laguna (1993) citados por GLAZAR (2000), os atributos de movimentos mais comuns são:

- alteração do valor de uma variável x_j de 0 para 1;
- alteração do valor de uma variável x_j de 1 para 0;
- as alterações de (1) e (2) simultaneamente, para dois diferentes valores de j ;
- alteração do valor da função objetivo de $c(x^a)$ para $c(x^b)$;
- alteração do valor de uma função $g(x^a)$ para $g(x^b)$ onde g pode representar uma função que ocorre naturalmente no problema, ou que é criada estrategicamente. Por exemplo, g pode ser uma medida de distância entre alguma solução dada, e uma solução referência, como a melhor solução até então encontrada.
- alteração no valor da diferença $g(x^b) - g(x^a)$;
- alterações combinadas dos dois últimos atributos, para mais de uma função g , consideradas simultaneamente.

4.7. Critérios de aspiração

Critérios de aspiração são introduzidos na busca tabu para determinar quando as regras de ativação tabu podem ser anuladas, removendo-se assim a classificação tabu aplicada ao movimento. O domínio do estado tabu dos movimentos através dos critérios de aspiração podem ser muito importantes para capacitar o método de BT a

alcançar seu melhor nível de performance, evitando que soluções não atrativas sejam visitadas.

Os principais critérios de aspiração são (GLOVER e LAGUNA, 1997):

- aspiração *default*: Em situações onde a vizinhança seja eventualmente pequena, ou quando a permanência tabu seja suficientemente grande é possível que uma iteração possa ocorrer quando todos movimentos disponíveis são classificados como tabu. Este critério permite que, nessas situações onde todos os movimentos possíveis são classificados de tabu e nenhum deles levar a uma melhora da solução, o movimento com a condição tabu mais antiga seja executado;
- aspiração por objetivo: Este critério permite que um movimento tabu seja executado se levar a uma solução melhor que a solução até então encontrada.

4.8. Oscilação estratégica

Oscilação estratégica está intimamente ligada à origem da BT, e produz uma forma de realizar uma interação efetiva entre intensificação e diversificação (GLOVER e LAGUNA, 1997). Segundo GLAZAR (2000), a oscilação estratégica é uma modificação no método das penalidades e representa uma outra forma de diversificação interessante. De acordo com GLOVER e LAGUNA (2000), a idéia desse método é oscilar entre regiões viáveis e inviáveis, manipulando a função objetivo, através de penalidades ou incentivos, ou alterando movimentos construtivos (melhora da função objetivo) e destrutivo (piora da função objetivo).

Segundo GLAZAR (2000), o uso da oscilação estratégica tem várias vantagens: permite executar movimentos menos complexos; o movimento direcionado para fora da fronteira de viabilidade e seu retorno em diferentes direções abre oportunidades para a melhorar as soluções que não seriam obtidas se a busca ficasse confinada; e finalmente, se o espaço de soluções viáveis é disjunto, então a oscilação estratégica provê um mecanismo de atravessar regiões de inviabilidade.

Gendrau et al., 1994; citados por GLAZAR (2000), propuseram o seguinte método de oscilação estratégica: para cada restrição i relaxada, uma penalidade pi é multiplicada ao excesso da violação desta restrição e adicionada a função objetivo de minimização. Esta penalidade é alterada a cada Lp iterações, utilizando-se pesos (g) através dos seguintes critérios:

- se todas as Lp soluções anteriores foram viáveis, então $pi = pi/g$
- se todas as Lp soluções anteriores foram inviáveis, então $pi = gpi$;
- caso contrário, não altere.
- segundo os mesmos autores, os valores mais usados para Lp e g são, respectivamente, de 10 e 2.

4.9. Classificação da busca tabu

Os algoritmos baseados na BT são divididos em duas classes, de acordo com a natureza do sistema, determinístico ou estocásticos, que por sua vez são subdivididos de acordo com algumas características de implementação (GLAZAR, 2000). A classificação apresentada no Quadro 1 e discutida a seguir foi extraída do trabalho de GLAZAR (2000).

Quadro 1 - Classificação dos algoritmos baseados na Busca Tabu

DETERMINÍSTICO	ESTOCÁSTICO
BT estrita (strict TS) BT fixada (fixed TS) BT reativa (reactive TS, RTS) BT paralela	BT probabilística BT robusta BT fixada com tie breaking estocástico BT reativa com tie breaking estocástico BT reativa com amostra da vizinhança (estratégia de lista de candidatos estocásticos)

em que: TS (Tabu Search).

- BT estrita: uma solução da vizinhança é proibida se, e somente se, ela já foi visitada no passado.
- BT fixada: um movimento é considerado tabu durante um intervalo de $|T|$ iterações.

- BT reativa: o tamanho $|T|$ da lista é ajustado automaticamente durante a pesquisa, identificando ciclos de soluções já visitadas.
- BT paralela: são realizados cálculos paralelos para a avaliação de diferentes movimentos.
- BT probabilística: as restrições tabu são substituídas por regras de probabilidade, onde baixas probabilidades proíbem o movimento.
- BT robusta: o tamanho $|T|$ da lista tabu é escolhido aleatoriamente dentre um valor máximo e um valor mínimo durante a pesquisa.
- BT reativa e BT fixada com tie breaking estocástico: desempate entre os melhores movimentos é feito através de regras aleatórias.
- BT reativa com amostra da vizinhança: quando a avaliação de toda a vizinhança é cara, uma amostra estocástica da vizinhança é avaliada através da estratégia de lista de candidatos estocástica.

5. SIMULATED ANNEALING

A metaheurística *Simulated Annealing* (SA) foi derivada de processos de recozimento sólidos (Metropolis *et al.*, 1953), citado por YOUSSEF, 2001). A analogia com a otimização combinatória foi introduzida por KIRKPATRICK *et al.* (1983) e aperfeiçoada por CERNY (1985). A SA pode ser utilizada na solução de uma variedade de difíceis problemas de otimização combinatória. O termo *annealing* refere-se a um processo térmico que começa pela liquefação de um cristal, a uma alta temperatura, seguido pela lenta e gradativa diminuição de sua temperatura, até que o ponto de solidificação seja atingido, quando o sistema atinge um estado de energia mínima (IGNACIO *et al.*, 2000).

A heurística SA é basicamente um algoritmo de busca local, sendo que a escolha de um novo elemento na vizinhança é realizada aleatoriamente. Os algoritmos de busca local apresentam o inconveniente de estacionar num ótimo local. A heurística SA evita este problema, utilizando uma probabilidade de aceitação de uma solução que piora a solução corrente, o que implica na

possibilidade de se escapar de um ótimo local. Esta probabilidade vai ficando menor, à medida que se aproxima da solução ótima (IGNACIO et al., 2000).

A heurística SA opera da seguinte forma: a partir de uma solução inicial x e através de um mecanismo de movimento m qualquer, uma nova solução x' é gerada. Após efetuado o movimento e calculado o valor da função objetivo da nova solução ($f(x')$), o movimento m será aceito se um dos seguintes critérios for satisfeito (BOSTON e BETTINGER, 1999):

- se o valor da função objetivo da nova solução " $f(x')$ " for melhor do que o valor da função objetivo da melhor solução encontrada até o momento, $f^*(x)$, o movimento m é aceito, caso contrário;
- se $f(x') < f^*(x)$, uma constante Z é calculada, e o movimento será aceito se o valor de Z for maior que um número (r) aleatório e contínuo, uniformemente distribuído entre 0 e 1 ($Z > r$), onde:

$$Z = e^{[f(x') - f^*(x)] / K_B \cdot T} \quad (8)$$

em que:

T = temperatura do sistema;

K_B = constante de *Boltzmann*;

e = constante.

A probabilidade (Z) para aceitação de uma solução de pior qualidade é conhecida como critério de *Metropolis* (Metropolis et al., 1983, citado por YOUSSEF, 2001).

O Algoritmo básico da heurística SA é apresentado a seguir, onde x é a solução inicial e T_0 é a temperatura inicial (Ribeiro, 1998 citado por IGNACIO et al., 2000).

Algoritmo *Simulated Annealing* (x, x', T);

(x é uma solução qualquer);

(x' é a solução vizinha de x);

(T é a temperatura);

Início

Enquanto temperatura (T) elevada **faça**

 Gerar x' de x ;

 Avaliar a variação de energia

$$\Delta E = f(x') - f(x)$$

Se $\Delta E > 0$ **então** $x \leftarrow x'$,

Caso contrário

Gerar $r \in \text{Unif}[0,1]$

Se $r < z = e^{(f(x') - f(x))/K \cdot T}$

Então $x \leftarrow x'$

Reduzir a temperatura T

Testar critério de parada

Fim.

De acordo com a expressão 8 acima, quanto menor a temperatura T , menor será a probabilidade (Z) de se aceitar soluções que piorem a função objetivo, o que se traduz em rejeitar cada vez mais esse tipo de solução. Quando T é maior, a probabilidade (Z) fica próxima de um, o que torna alta a probabilidade de uma solução de pior qualidade ser aceita. Segundo IGNACIO et al. (2000), em geral, a constante de *Boltzmann* (K_B) não é considerada em problemas de otimização.

A habilidade para evitar paradas prematuras em ótimos locais com o uso de variáveis aleatórias, onde um movimento m com uma solução de pior qualidade pode ser aceito com uma probabilidade Z , é o principal componente da *simulated annealing*. À medida que o algoritmo percorre o espaço de solução, a temperatura vai sendo diminuída e a probabilidade de aceitação de uma solução inferior vai se tornando cada vez menor.

A temperatura (T) é considerada como um parâmetro de controle da mobilidade do sistema, ou seja, traduz a flexibilidade de aceitar mudanças no valor da função de custo. A probabilidade de se aceitar movimentos de pior qualidade decresce à medida que T diminui. Em temperaturas altas a busca é mais aleatória, enquanto para temperatura menores o processo de busca torna-se mais guloso, ou seja, diminui a chance de um movimento de pior qualidade ser aceito. Em temperaturas nulas, a busca torna-se totalmente gulosa, i.é., apenas movimentos de boa qualidade são aceitos (Kirkpatrick et al., 1983, citados por YOUSSEF, 2001).

A temperatura (T) da SA em um dado instante é função da temperatura inicial (T_0) e da sua taxa de redução (T_R) e do número de interação na qual uma dada temperatura é mantida constante nT_{const} . Esses parâmetros são críticos para o sucesso do algoritmo *simulated annealing*. Existem atualmente poucas diretrizes

para determinar os parâmetros apropriados para *simulated annealing*. Esses valores, na maioria das vezes, dependem da natureza do problema, devendo ser determinados por experimentação (BOSTON e BETTINGER, 1999). Segundo IGNACIO et al., (2000), muitos autores tentam encontrar programas de redução ótimos, motivados pela diminuição do tempo da CPU. Segundo estes mesmos autores, ao se considerar temperaturas iniciais muito elevadas, a qualidade da solução final da SA passa a depender menos da qualidade da solução inicial, já que qualquer que seja essa solução, a sua probabilidade de aceitação será alta. Entretanto, a escolha de uma temperatura inicial alta exigirá um maior corrida do algoritmo até a sua convergência, onerando o custo de processamento. Em geral o valor da temperatura inicial é dependente do problema (IGNACIO et al., (2000). O algoritmo SA necessita iniciar com uma temperatura (T) alta. Entretanto, se esse valor inicial de T é muito alto, pode haver uma piora no tempo de processamento. O valor da temperatura inicial deve ser tal que permita virtualmente que ambos os movimentos de melhor ou pior qualidade sejam aceitos (inicialmente, a probabilidade de aceitação dos movimentos, bons ou ruins devem ser aproximadas). YOUSSEF et al. (2001) utilizaram um procedimento descrito por Wong e Liu em 1986 para calcular a temperatura inicial, baseado no procedimento de *Metropolis*. O procedimento executa um número finito de passos (M) do algoritmo SA, calculando os respectivos valores das funções de custo (f_i) de cada um desses movimentos. Uma diferença na função de custo de cada movimento i (dado por $\Delta f_i = f_i - f_{i-1}$) é calculada. Denominado por M_u e M_d o número de movimentos de melhora e piora, respectivamente (onde $M = M_u + M_d$), calcula-se a variação média de energia (ΔF_{Med}) utilizando a expressão:

$$\Delta F_{Med} = \frac{1}{M_d} \sum_{i=1}^{M_d} \Delta f_{imes i} \quad (9)$$

Para manter a probabilidade de aceitação (P_0) de movimentos de melhora ou piora aproximadamente iguais no estágio inicial da SA, estima-se o valor do parâmetro temperatura substituindo o valor de P_0 na expressão seguinte, derivada da função de *Metropolis*:

$$T_0 = \frac{\Delta fitness_d}{\ln(P_0)} \quad (10)$$

em que:

$$P_0 \approx 1 \quad (P_0 = 0.999).$$

Da mesma forma, o parâmetro T_R pode afetar a performance da heurística SA. Em geral, quanto menor o valor desse parâmetro, maior será o tempo de convergência do algoritmo, uma vez que mais soluções necessitarão ser visitadas para que haja maiores chances de encontrar uma solução de maior qualidade. Algumas vezes um processo lento de redução da temperatura pode ser necessário para se obter soluções competitivas, implicando em maiores tempos de processamento (IGNACIO et al., (2000).

Uma das vantagens da SA é o fato de que, em cada iteração, uma única solução é visitada, ao contrário de muitas técnicas heurística como os AGs e a BT, onde várias soluções são visitadas em cada iteração. Desta forma, processamentos como cálculo da função objetivo e outros processamentos necessários para avaliar as soluções pesquisadas não degradam muito a eficiência do algoritmo (IGNACIO et al., 2000). Uma das desvantagens da SA é o fato de a heurística utilizar poucas informações do problema. De modo geral, a SA utiliza somente a variação do valor da função objetivo nas avaliações dos movimentos, o que a torna pouco “inteligente” (não se consideram outros aspectos caraterísticos do problema na pesquisa do espaço de soluções).

5. MÉTODOS METAHEURÍSTICOS HÍBRIDOS

Uma idéia interessante que tem despertado crescente atenção dos pesquisadores refere-se ao desenvolvimento de métodos heurísticos híbridos utilizando as metaheurísticas Busca Tabu, *Simulated Annealing* e Algoritmo Genético. O objetivo dos modelos híbridos é combinar as técnicas, preservando suas características individuais de ação *inteligente*, de tal forma que o procedimento resultante seja mais eficaz do que qualquer um dos seus componentes isoladamente (SOUZA e MOCCELLIN, 2000). Estes autores citam diversos exemplos de

aplicações dessas técnicas para problemas da classe planejamento e controle da produção.

6. MEDIDAS DE PERFORMANCE

Duas medidas são particularmente recomendadas para teste de desempenho de heurísticas, e são elas: medidas de eficácia e medidas de eficiência. Segundo Chiavenato (1987), citado por NOBRE (1999), a eficácia é uma medida de alcance de resultados, enquanto que eficiência é uma medida de utilização de recursos disponíveis durante o processo. Desta forma, a eficácia da heurística pode ser entendida como uma medida que representa a distância entre o resultado obtido e o melhor resultado possível de ser encontrado por outros métodos. Por outro lado, a eficiência da heurística pode ser vista como uma medida dos recursos computacionais gastos para alcançar os resultados.

De acordo com BOSTON e BETTINGER, 1999, existem basicamente quatro formas de se comparar a eficácia de uma heurística:

- comparar os seus resultados com soluções ótimas conhecidas, através de soluções do mesmo problema por algoritmos exatos¹;
- comparar os resultados com problemas onde as restrições inteiras são relaxadas, neste caso, utiliza-se algoritmos exatos como *simplex* .
- comparar seus resultados com outras heurísticas, o que permite apenas comparações relativas entre as técnicas.
- estimativas da solução ótima através de procedimentos estatísticos, utilizando-se técnicas do valor extremo.

As medidas de eficiência, por outro lado, são mais difíceis de se obter, envolvendo análise de complexidade de algoritmos como complexidade de tempo e de espaço. Entretanto, muitos trabalhos empregam medidas mais simples como o tempo computacional e o número de iterações gastos para obter uma solução para o problema.

¹ Este é o método ideal, mas pode ser inviável para muitos problemas devido ao custo para obtenção da solução ótima.

7. APLICAÇÕES NO MANEJO FLORESTAL

As heurísticas modernas têm despertado crescente interesse da comunidade científica em virtude da qualidade das soluções encontradas, tornando-se uma alternativa cada vez mais interessante para a solução de grande parte das aplicações reais dos modelos combinatórios (NOBRE, 1999).

A maioria das propostas heurísticas na área florestal foram apresentadas para solucionar problemas com restrições de adjacência (RODRIGUEZ, 1994) e de singularidade, uma vez que tais restrições aumentam muito o tamanho do problema e impõem o uso de variáveis inteiras. Assim como em outras áreas da ciência, os primeiros estudos com aplicações heurísticas na área florestal utilizaram técnicas heurísticas específicas para resolver determinados problemas. As pesquisas evoluíram mais tarde enfocando uso das chamadas metaheurísticas para resolver problemas semelhantes. A seguir é feita uma síntese, sem a intenção de esgotar todos os trabalhos publicados na área, de alguns estudos envolvendo a utilização das técnicas heurísticas.

Formulações de problemas envolvendo restrições espaciais como as restrições de adjacência e de habitat para animais silvestres ganharam grande atenção nos últimos anos. Um problema comum na formulação dessas restrições é a ocorrência de restrições redundantes. YOSHIMOTO e BRODIE (1994a) desenvolveram uma heurística para eliminar esses problemas e comparou-a com duas outras técnicas de formulação.

WEINTRAUB et al. (1994) desenvolveram o algoritmo *column generation* baseado em programação linear para resolver problemas de adjacência em manejo de recursos florestais. O algoritmo utiliza uma heurística para arredondar as soluções não inteiras e foi aplicado a problemas de porte médio, propiciando resultados satisfatórios em termos de esforço computacional.

MURRAY e CHURCH (1996) apresentam modelos de decisão de colheita florestal utilizando variáveis inteiras para formular problema com restrições de singularidade (adotar uma única alternativa de manejo em cada unidade de manejo)

e restrições de adjacência e reforçaram a crença de que as técnicas heurísticas podem ter muito sucesso em encontrar soluções boas para esses problemas.

Rodriguez (1994), citado por NOBRE (1999) apresentou a heurística da razão-R com bons resultados tanto em termos de comparação com a solução ótima matemática quanto em termos de processamento. Em um dos testes envolvendo 2584 variáveis inteiras, e utilizando microcomputadores com processadores de geração anterior à linha Pentium, a sua heurística encontrou soluções em aproximadamente 12 minutos. Tais testes mostram a potencialidade de sua heurística para a solução de grandes problemas de planejamento florestal. Posteriormente RODRIGUEZ e McTAGUE (1997) e NOBRE (1999) apresentaram melhorias na heurística e novos testes.

BORGES et al. (1999) utilizaram uma heurística baseada em programação dinâmica para resolver problemas de planejamento de colheita em povoamentos de eucalipto com restrições de adjacência. Apesar dos bons resultados produzidos pela heurística, uma das críticas do modelo é sua incapacidade de contemplar importantes restrições para o gerenciamento de recursos florestais, tais como: fluxo de madeira, renda periódica, áreas de plantio etc.

As metaheurísticas "*Interchange*", "*Tabu Search*" e "*Simulated Annealing*" foram aplicadas por MURRAY e CHURCH (1995) para resolver problemas de colheita florestal incluindo restrições de transporte, adjacência e financeiras. Os autores apresentam resultados nos quais as heurísticas alcançaram 90% da solução ótima.

Brumelle et al. (1996) citado por GLOVER e LAGUNA (1997) formulou um problema de planejamento florestal envolvendo o manejo de bacias hidrográficas com múltiplos objetivos e restrições espaciais. O problema foi formulado como um problema de otimização discreta. O estudo considerou dois problemas constituídos por 219 e 491 unidades de corte (talhões), onde foi utilizada a metaheurística Busca Tabu para resolvê-los.

Lokketangen e Hasle (1997) citados por GLOVER e LAGUNA (1997), mostraram como um problema de planejamento de colheita florestal pode ser resolvido usando técnicas baseadas em BT. A busca na vizinhança consistiu de

dois níveis onde em um primeiro nível seleciona-se um povoamento para efetuar um movimento, e num segundo nível seleciona-se uma alternativa de manejo para este povoamento. O conceito de permanência tabu localizada foi aplicado na seleção do povoamento.

LAROZE (1999) utilizou a heurística Busca Tabu para otimização da colheita e desdobramento das toras em tábuas, incluindo restrições de mercado quanto à qualidade das toras. Foram alcançados valores de 95% da solução ótima matemática para todos os cenários analisados.

Bettinger et al. (1997), citados por BOSTON e BETTINGER (1999), utilizaram a Busca Tabu para resolver problemas de planejamento florestal, envolvendo restrições espaciais para habitat de *Cervus elephus*.

BOSTON e BETTINGER (1999), utilizaram a BT para resolver problemas de planejamento florestal com restrição de singularidade e adjacência e comparou-a com duas outras técnicas heurísticas: Programação Inteira Monte Carlo e *Simulated Annealing*. Foram geradas quinhentas soluções para cada um dos quatro problemas que possuíam entre 3000 e 5000 variáveis inteiras 0-1, *Simulated Annealing* foi a técnica que encontrou o maior valor (cerca de 99%) para a função objetivo para três dos quatro problemas, enquanto BT encontrou a melhor solução para um dos problemas. Entretanto, quando se mediu a qualidade da solução dada pela variância das quinhentas soluções geradas, BT foi a técnica que apresentou a menor variação, portanto, com melhores resultados.

Outras aplicações de técnicas heurísticas em planejamento de recursos florestais podem ser encontradas em MENDOZA (1985), TORRES-ROJO (1990), LOCKWOOD e MORE (1993), LAROZE e GLEBER (1997) e HOGANSON e BORGES (1998).

8. CONCLUSÕES

Os objetivos deste capítulo foram apresentar os fundamentos e algumas aplicações das metaheurísticas Algoritmo Genético, Busca Tabu e *Simulated Annealing* com ênfase no gerenciamento florestal.

Os principais trabalhos envolvendo a utilização de algoritmos exatos como o algoritmo *branch and bound* para solução de problemas combinatorias mostraram que estes são eficientes apenas em problemas de pequeno porte.

Por outro lado, muitos problemas de planejamento florestal são de natureza combinatorial e podem envolver centenas a milhares de variáveis e não podem ser resolvidos eficientemente por tais métodos.

As técnicas heurísticas têm tido bastante sucesso na solução de problemas combinatoriais em diversas áreas, e alguns estudos já foram desenvolvidos na área florestal com sucesso.

Existem muitas técnicas heurísticas disponíveis para solução de problemas combinatoriais, indo desde heurísticas menos convencionais até as chamadas metaheurísticas. As metaheurísticas mais estudadas atualmente têm sido a Busca Tabu, *Simulated Annealing* e Algoritmos Genéticos.

Neste capítulo foi apresentada uma revisão bibliográfica abordando os princípios básicos e algumas aplicações das três metaheurísticas: Busca Tabu, *Simulated Annealing* e Algoritmos Genéticos. Para ilustrar os conceitos discutidos neste capítulo, nos capítulos que se seguem serão apresentadas aplicações das metaheurísticas Algoritmo Genético, Busca Tabu e *Simulated Annealing*.

Com base na revisão bibliográfica pôde-se concluir que as metaheurísticas Algoritmos Genéticos, Busca Tabu e *Simulated Annealing* apresentam aplicações para solução de variados problemas, em especial os problemas de otimização combinatoria, problemas esses de difícil solução através dos algoritmos exatos tradicionais. Considerando que muitos problemas de gerenciamento florestal são de natureza combinatorial, a aplicação dessas metaheurísticas têm-se estendido para este setor, com algumas pesquisas já desenvolvidas.

9. REFERÊNCIAS BIBLIOGRÁFICAS

BORGES, J.G. Paradigmas, tecnologias e equívocos em gestão de recursos florestais. **Revista Florestal**, Lisboa, 12(1), p.27-34, 1999.

BORGES, J.G.; GOMES, J.A.; FALCÃO, A.; HOGANSON, H.M. Heurística baseada em programação dinâmica e restrições de adjacência em gestão florestal. **Revista Florestal**, Lisboa, 12(1), p.99-107, 1999.

BOSTON, K.; BETTINGER, P. An Analysis of Monte Carlo Integer Programming, Simulated Annealing, and Tabu Search Heuristics for Solving Spatial Harvest Scheduling Problems. *Forest Science, Washington*, 45(2):292-301, 1999.

CERNY, V. Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1): 41-51, 1985.

GARCIA, A.L.; GARCIA, B.; FRIEDMANN, C.V. Resolução de um problema de equilíbrio de trabalho e distribuição de agentes de serviço. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p923-935.

GLAZAR, E.J. **Análise de diferentes estratégias na aplicação da busca tabu ao problema de roteamento de veículos.** Rio de Janeiro, RJ: COPPE/UFRJ, 2000. 132p. Dissertação (Mestrado em Ciências em Engenharia de Produção - Universidade Federal do Rio de Janeiro, 2000.

GLOVER, F. Heuristics for integer programming using surrogate constraints. **Decision Science**, 8, 156-166. 1977.

GLOVER, F. Tabu Search - Part I, ORSA. **Journal on Computing** 1, 190-206. 1989.

GLOVER, F.; LAGUNA, M. **Tabu search.** Kluwer academic publishers, Boston. 1990. 382p.

GLOVER, F.; LAGUNA, M. **Tabu search.** Kluwer academic publishers, Boston. 1997. 382p.

GOLDBARG, M.C.; LUNA, H.P.L. Otimização combinatória e programação linear: modelos e algoritmos. **Rio de Janeiro, Campus, 2000. 649p.**

GOLDBARG, M.C.; GOUVÊA, E.F. Transgenética computacional. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p1537-1569.

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

HOGANSON, H.M.; BORGES, J.G. Using Dynamic Programming and overlapping subproblems to address adjacency in large harvest scheduling problems. **Forest Science**, Washington, 44: p526-538, 1998.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**, Ann Harbor, University of Michigan Press, 1975.

IGNACIO, A.A.V.; FERREIRA FILHO, V.J.M.; GALVÃO, R.D. Métodos heurísticos num entorno paralelo. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p769-788.

IGNIZIO, J.P.; CAVALIER.T.M. **Linear Programming**. Englewood Cliffs: Prentice Hall, 1994. 666p.

KIRKPATRICK Jr., S., GELATT, C., VECCHI, M. Optimization by simulated annealing. *Decision Science*, **220(4598)**, 498-516, 1983.

LAROZE, A.J. A linear programming, tabu search method for solving forest-level bucking optimization problems. **Forest Science**, Washington, 45(1):108-116, 1999.

LAROZE, A.J.; GREBER, B.J. Using Tabu search to generate stand-level, rule-based bucking patterns. **Forest Science**, Washington, 43(2):157-169, 1997.

LOCKWOOD, C.; MOORE, T. Harvest scheduling with spatial constraints: a simulated annealing approach. **Can. J. For. Res.**, 23: 468-478, 1993.

MENDOZA, G.A. A heuristic programming approach in estimating efficient target levels in goal programming. **Can J. For. Res.** 16: 363-366, 1985.

MITCHELL, M. An introduction to genetic algorithms. **London, A Bradford Book, 1996. 205 p.**

MURRAY, A.T. Spatial restrictions in harvest scheduling. **Forest Science**, Washington, 45(1): 45-52, 1999.

MURRAY, A.T.; CHURCH, R.L. Analyzing Cliques for imposing adjacency restrictions in forest models. **Forest Science**, Washington, 42(2), p.45-52, 1996.

NOBRE, S.R. **A heurística da Razão-R aplicada a problemas de gestão florestal**. Lavras, MG: UFLA, 1999. 128p. Dissertação (Mestrado em Administração Rural) - Universidade Federal de Lavras, 1999.

REBELLO, F.R.; HUMACHER, S. Uma proposta de algoritmo genético de duas fases para roteamento de veículos. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000.

RODRIGUEZ, L.C.E. **Use of heuristic methods in the analysis of large-scale forest management integer programming models**. Athens-Georgia: University of Georgia, 1994. 93p. (Tese de doutorado).

RODRIGUEZ, L.C.E.; MCTAGUE, J.P. Integer solutions for model I harvest scheduling LP problems: testing the R-ratio heuristic. In: SYPOSIUM ON SYSTEMS ANALYSIS IN FOREST RESOURCES, 7, Bellaire, 1997.

ROMERO, R.; GALLEGOS, R.A. Algoritmo genético especializado para problemas de alocação ótima de capacitores em sistemas de distribuição de energia elétrica. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p835-847.

SOUZA, A.B.D.; MOCCELLIN, J.V. Metaheurística híbrida algoritmo genético-busca tabu para programação de operações flow shop. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p314-324.

TORRES-ROJO, J.M., BRODIE, J.D. Adjacency constraints in harvest scheduling: na aggregation heuristic. **Can J. For. Res.**, 20: 978-986, 1990.

- WEINTRAUB, A.; JONES, G.; MAGENDZO, A.; MEACHAM, M.; KIRBY, M. A heuristic system to solve mixed integer forest planning models. **Oper. Res.** 42 (6): 1010-1024, 1994.
- YAMASHITA, D.S.; MAZZINI, R. ARMENTANO, V.A. Busca Tabu aplicada à programação de tarefas em máquinas paralelas com tempos de preparação dependentes da sequência e datas de entrega. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p389-396.
- YOSHIMOTO, A.; BRODIE, D. Comparative analysis of algorithms to generate adjacency constraints. **Can. J. For. Res.**, 24: 1277-1288, 1993.
- YOUSSEF, H.; SAIT, S.M.; ADICHE, H. Evolutionary algorithms, simulated annealing and tabu search: a comparative study. **Engineering Applications of Artificial Intelligence**. 14: 167-181, 2001.
- WERRA, D. HERTZ, A. Tabu search techniques: a tutorial and an application to neural networks. **OR Spektrum**, v.11, 131-141, 1989.

CAPÍTULO 4

METAHEURÍSTICA ALGORITMO GENÉTICO PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL COM RESTRIÇÕES DE INTEIREZA

RESUMO - Este capítulo teve como objetivo desenvolver e testar um Algoritmo Genético (AG), para a solução de problemas de gerenciamento florestal com restrições de inteireza. O AG foi testado em cinco problemas, contendo entre 12 e 423 variáveis de decisão, sujeitos às restrições de singularidade, produção mínima e produção máxima periodicamente. Todos os problemas tiveram como objetivo a maximização do valor presente líquido. O AG foi codificado em ambiente *delphi 5.0* e os testes efetuados em um microcomputador AMD K6II 500 MHz, com memória RAM de 64 MB e disco rígido de 15GB. A performance do AG foi avaliada de acordo com as medidas de eficácia e eficiência. Os diferentes valores ou categorias dos parâmetros do AG foram testados e comparados quanto aos seus efeitos na eficácia do algoritmo. A seleção da melhor configuração de parâmetros foi feita utilizando o teste L&O, a um nível de 1% de probabilidade e análises através de estatísticas descritivas. A melhor configuração de parâmetros propiciou ao AG, eficácia média de 94,28%, valor mínimo igual a 90,01%, valor máximo igual a 98,48%, com coeficiente de variação de 2,08% do ótimo matemático, obtido pelo algoritmo exato *branch and bound*. Para o problema de maior porte, a eficiência do AG foi cinco vezes superior à eficiência do algoritmo exato *branch and bound*. O AG apresentou-se como uma nova abordagem bastante atrativa para solução

de importantes problemas de gerenciamento florestal de difícil solução, ou até mesmo, impossível de serem solucionados através do uso dos instrumentos computacionais da atualidade.

Palavras-chave: Gerenciamento Florestal, Heurísticas, Algoritmos Genéticos.

METAHEURISTIC GENETIC ALGORITHMIC TO SOLVE FOREST PLANNING PROBLEM WITH INTEGER CONSTRAINTS

ABSTRACT – The objective of his chapter was to develop and test a Genetic Algorithm (GA) to solve problems of forest management with integer constraints. GA was tested in five problems containing between 12 and 423 decision variables, subjected to singularity constraints, minimum and maximum production periodically. All the problems had the objective of maximizing the net present value. GA was codified into *delphi* 5.0 language and the tests were performed in a microcomputer AMD K6II 500 MHZ, with RAM memory of 64 MB and hard disk of 15GB. The GA performance was evaluated according to the efficacy and efficiency measures. The different values or categories for the GA parameters were tested and compared in relation to their effects on the algorithm efficacy. The selection of the parameters' best configuration was performed by using the L&O test at 1% probability and analyses via descriptive statistics. The parameters' best configuration provided for GA average efficacy of 94.28%, minimum value equal to 90.01%, maximum value equal to 98.48%, with coefficient of variation being 2.08% of the mathematical optimum, obtained by the exact algorithm *branch and bound*. As for the larger problem, the efficiency of GA was five times superior to the efficiency of the exact algorithm *branch and bound*. GA came out as a quite attractive new approach in forest management for solving important problems of difficult solution or even impossible of being solved through the use of the current computational instruments.

Key- words: Forest Management, Heuristic and Genetic Algorithm.

1. INTRODUÇÃO

Os Algoritmos Genéticos (AGs) são técnicas de otimização robustas e efetivas inspiradas por mecanismos de evolução natural e genética (HOLLAND, 1975 e GOLDBERG, 1989;) e têm sido utilizados na solução dos mais variados problemas de diferentes domínios, como programação automática, ecologia, Pesquisa Operacional etc (GARCIA et al., 2000). Para REBELLO e HAMACHER (2000), os AGs são métodos robustos que podem ser utilizados para resolver problemas em pesquisa numérica, otimização de funções e aprendizagem de máquina, entre outras aplicações. Eles têm sido apontados como técnicas promissoras para resolver problemas combinatoriais, como os clássicos problemas de Programação Inteira. Apesar de não garantirem otimalidade, possuem como vantagem principal um melhor desempenho ou performance computacional, em relação aos métodos concorrentes, além de ser um procedimento de relativa simplicidade (GARCIA et al., 2000). Segundo estes autores, praticamente todos os problemas NP-completos ou NP-difíceis têm uma versão de solução heurística que usa esse tipo de algoritmo.

Muitos problemas de planejamento florestal apresentam características próprias para solução mais eficiente por meio de AG. Exemplos de problemas combinatoriais em planejamento florestal são: regulação florestal com escolha de um único regime por talhão, problema de adjacências em talhões selecionados para colheita, seleção de rota de veículos, seccionamento de toras, corte em indústria moveleira e corte de papel, dentre outros. Desta forma, os AGs podem ser uma ferramenta com grande potencial que necessita ser investigado, principalmente pelo fato de não ter sido observado até então, nenhum trabalho reportando a aplicação desses algoritmos na solução de problemas do setor florestal. O estudo apresentado neste capítulo teve os seguintes objetivos:

- Desenvolver e testar um AG para resolver problemas de planejamento florestal;
- Ilustrar aplicação do procedimento em um problema de pequeno porte;
- Desenvolver um código computacional para possibilitar o teste e comparação dos possíveis efeitos de alguns de seus parâmetros na estratégia de solução em alguns problemas;

2. MATERIAIS E MÉTODOS

2.1. Descrição dos problemas

Para testar o AG desenvolvido foram utilizados cinco problemas de planejamento florestal, conforme descritos no Quadro 1. O problema 1 foi utilizado para ilustrar uma corrida do AG. Os problemas 2, 3, 4 e 5, apresentando algumas variações em termos de número de variáveis e número e magnitude das restrições de produção (demanda) mínimas e máximas, foram utilizados para testar a performance do AG.

Quadro 1 – Descrição geral dos problemas escolhidos para aplicação do AG, formulados através do sistema de planejamento florestal SisFlor.

Problema	Nº de unidades de manejo	Nº variáveis de decisão	Horizonte de planejamento	Restrição de produção mínima (m ³)	Restrição de produção máxima (m ³)
1	4	12	4	10000,00	-
2	10	93	10	15000,00	60000,00
3	10	206	15	15000,00	60000,00
4	19	215	10	15000,00	60000,00
5	40	423	10	30000,00	120000,00

Além das restrições de produção mínimas e máximas, todos os problemas tiveram como objetivos maximizar o valor presente líquido sujeito às restrições de singularidade (adotar um único regime de manejo em cada talhão).

2.2. Formulação dos problemas

Os problemas foram formulados utilizando-se o Sistema de Planejamento Florestal (SisFlor). A abordagem de formulação utilizada pelo SisFlor segue a estrutura do modelo I proposta por JOHNSON e SCHEURMAN (1977). A formulação matemática desse modelo para os problemas propostos é descrita a seguir.

- **Função objetivo**

Uma vez que o objetivo do problema é a maximização do valor presente líquido, a função objetivo do modelo é dada por:

$$Max . Z = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} \quad (1)$$

em que

X_{ij} = variável de decisão que representa a j -ésima alternativa de manejo adotada na i -ésima unidade de manejo.

C_{ij} = é o valor presente líquido total (\$) de cada unidade de manejo i , colhida segundo a alternativa j ;

M = número total de unidades de manejo;

N = número total de alternativas de manejo na i -ésima unidade de manejo.

- **Restrição de singularidade**

A imposição dessa restrição requer a formulação do problema seguindo a abordagem de formulação via Programação Inteira (PI), uma vez as variáveis de decisão são de natureza discreta, conforme mostrado a seguir:

$$\sum_{i=1}^M \sum_{j=1}^N X_{ij} = 1; \quad i = \{1, 2, \dots, M\} \quad (2)$$

$$X_{ij} = \begin{cases} 1, & \text{se a prescrição } j \text{ for adotada no talhão } i \\ 0, & \text{caso contrário} \end{cases}$$

- **Restrição de demanda mínima**

Esta restrição deve assegurar a alocação de uma produção para atender à uma demanda mínima preestabelecida, assim:

$$\sum_{i=1}^M \sum_{j=1}^N V_{ijk} X_{ij} \geq Dmin_k \quad \{k = 0, 1, \dots, H - 1\} \quad (3)$$

em que

V_{ijk} = volume (m^3) produzido pela i -ésima unidade de manejo quando a j -ésima alternativa de manejo é adotada, no início do período k ;

$Dmin_k$ = demanda mínima de madeira (m^3) em cada período (k) do horizonte de planejamento;

- **Restrição de demanda máxima**

Esta restrição assegura a alocação de níveis máximos de produção (demanda) de madeira em cada período do horizonte de planejamento, assim:

$$\sum_{i=1}^M \sum_{j=1}^N V_{ijk} X_{ij} \leq D \max_k \{k = 0, 1, \dots, H - 1\} \quad (4)$$

$Dmax_k$ = demanda máxima de madeira (m^3) em cada período (k) do horizonte de planejamento.

Considerando que o problema 1 foi utilizado como base para descrição dos princípios básicos da heurística AG e as demais heurísticas apresentadas nos capítulos seguintes, optou-se por apresentar mais detalhes desse problema, conforme mostrado no Quadro 2. As prescrições representam seqüências preestabelecidas de ações que ocorrem ao longo do horizonte de planejamento. Para cada unidade de manejo i , foram geradas três alternativas de manejo j . Para a situação em pauta, essas alternativas referem-se aos períodos de corte de uma dada área, seguidas de regeneração. O horizonte foi subdividido em períodos anuais e as ações ocorrem no início dos mesmos.

Os valores apresentados na área hachurada do Quadro 2 representam o volume total (m^3), obtido com a colheita da unidade de manejo i , através da opção j , no k -ésimo período (V_{ijk}).

Cada opção de manejo (X_{ij}) deve ter o seu respectivo coeficiente (C_{ij}) calculado, sendo C_{ij} o valor presente líquido, que é uma medida da diferença entre as receitas e os custos, atualizados por uma taxa de juros r .

Quadro 2 - Prescrições viáveis de manejo para o problema 1, considerando rotações iguais a cinco, seis e sete anos em um horizonte de planejamento de quatro anos subdividido em períodos anuais com o primeiro período igual a zero.

Unidade de manejo (i)	Área (ha)	Idade (anos)	Rotação (anos)	Opção	Horizonte de planejamento (H)				Receita líquida ¹ (C _{ij}) (\$)
					0	1	2	3	
1	40	5	5	X ₁₁	8000	-	-	-	40000
			6	X ₁₂	-	9600	-	-	60000
			7	X ₁₃	-	-	10800	-	80000
2	60	4	5	X ₂₁	-	8400	-	-	48000
			6	X ₂₂	-	-	12000	-	66000
			7	X ₂₃	-	-	-	12000	78000
3	50	5	5	X ₃₁	10000	-	-	-	60000
			6	X ₃₂	-	12000	-	-	75000
			7	X ₃₃	-	-	13500	-	90000
4	80	4	5	X ₄₁	-	19200	-	-	120000
			6	X ₄₂	-	-	21000	-	160000
			7	X ₄₃	-	-	-	24000	200000

Produção	colheita de madeira
-	nenhuma atividade de colheita

O modelo matemático para o problema 1, com base nas informações dos Quadros 1 e 2 anteriores, apresenta a seguinte estrutura:

Objetivo:

$$\begin{aligned} \text{Max } Z = & 40000 X_{11} + 60000 X_{12} + 80000 X_{13} + 48000 X_{21} + 66000 X_{22} + 78000 X_{23} \\ & + 60000 X_{31} + 75000 X_{32} + 90000 X_{33} + 120000 X_{41} + 160000 X_{42} + \\ & 200000 X_{43} \end{aligned}$$

Sujeito a:

- *Restrição de Singularidade*

$$X_{11} + X_{12} + X_{13} = 1$$

$$X_{21} + X_{22} + X_{23} = 1$$

$$X_{31} + X_{32} + X_{33} = 1$$

$$X_{ij} \in \{0,1\}$$

- *Restrição de produção mínima*

$$8000 X_{11} + 10000 X_{31} \geq 10000$$

$$9600 X_{12} + 8400 X_{21} + 12000 X_{32} + 19200 X_{41} \geq 10000$$

$$10800 X_{13} + 12000 X_{22} + 13500 X_{33} + 21000 X_{42} \geq 10000$$

¹ Valores hipotéticos utilizados para efeito de ilustração

$$12000 X_{23} + 24000 X_{43} \geq 10000$$

2.3. Desenvolvimento do AG

2.3.1. Codificação computacional do AG

O AG foi implementado dentro da rotina computacional denominada *SMAP-F* (Soluções Metaheurísticas Aplicadas ao Planejamento Florestal), desenvolvida utilizando o ambiente de programação *delphi 5.0*. Os testes com o algoritmo foram efetuados em um microcomputador AMD K6II 500 MHZ, com memória RAM de 64 MB e disco rígido de 15 GB. A estrutura geral do programa é apresentada na Figura1 e os detalhes da implementação descritas nos itens subseqüentes.

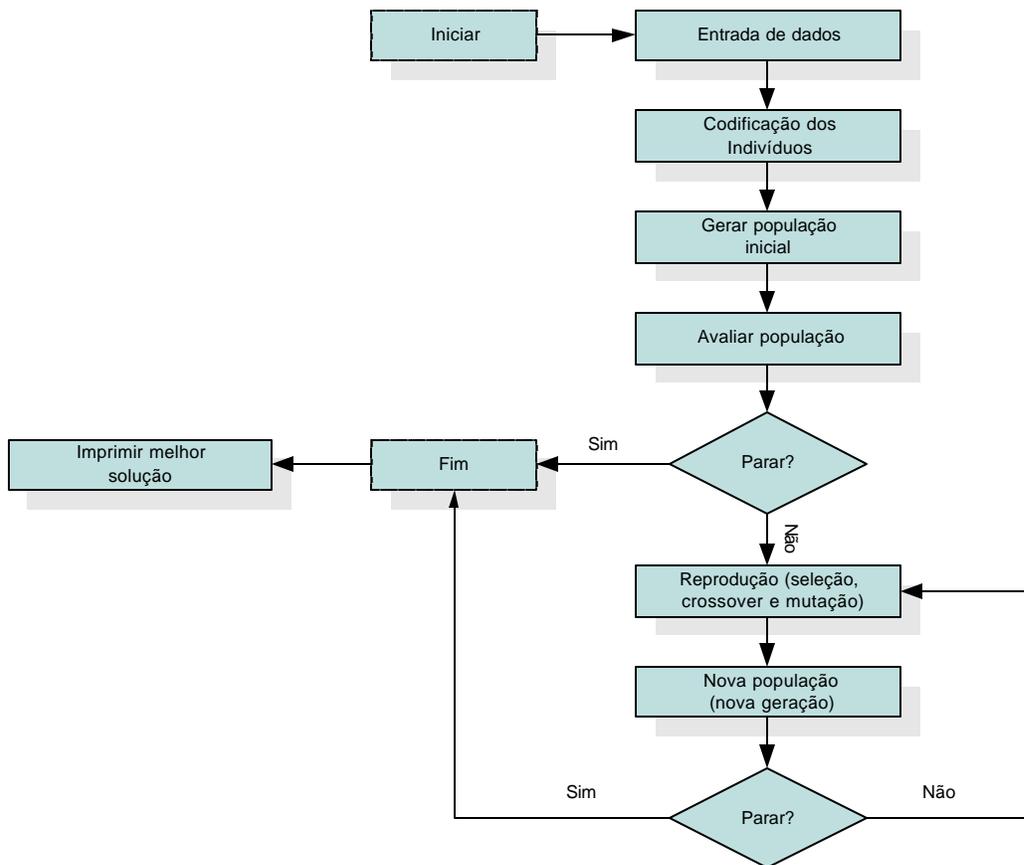


Figura 1 – *SMAP-F*: esquema geral do programa para implementação do AG.

2.3.2. Entrada de dados

O modelo matemático formulado através do sistema de planejamento SisFlor é gravado em um arquivo no formato texto (*txt*) e lido diretamente pelo *SMAP-F* para posterior alimentação do AG. O modelo matemático do arquivo texto importado possui a estrutura do modelo de Programação Linear, porém, pode ser convertido em um modelo de Programação Inteira. Os botões que permitem esta conversão podem ser visto no canto inferior esquerdo da Figura 2.

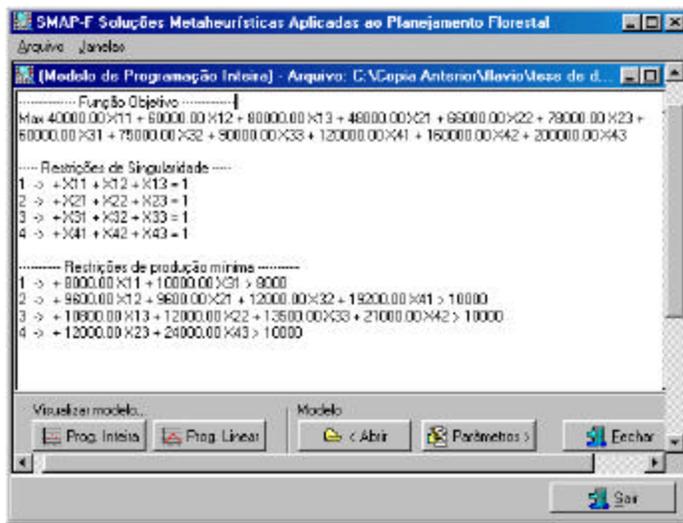


Figura 2– *SMAP-F* :interface para entrada de dados.

Além do modelo para representar o problema a ser resolvido, o AG requer como entrada de dados, a definição dos valores dos seus diferentes parâmetros. Este valores são definidos pelo usuário em um módulo específico, conforme discutido no item 2.3.10 desse capítulo.

2.3.3. Codificação dos indivíduos

Antes de apresentar a codificação dos indivíduos para o AG, é importante visualizar um esquema para representar uma solução para o problema em foco. Conforme já demonstrado, para representar as alternativas de manejo do problema foi utilizado uma variável de decisão X_{ij} simbolizando a alternativa de manejo j (j

$=1,2,\dots,N$) assinalada à unidade de manejo i ($i = 1,2,\dots,M$). Um vetor $V(x) = \{X_{11}, X_{12}, \dots, X_{ij}\}$, com estas respectivas variáveis de decisão foi esquematizado da seguinte forma (Figura 3):

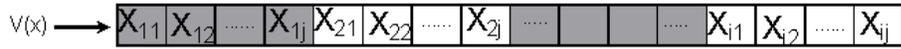


Figura 3 – Representação de um vetor com as variáveis de decisão do problema.

Uma solução possível (x) para o problema formulado através do modelo de programação inteira com as variáveis de decisão na forma binária ($X_{ij} \in \{0,1\}$), avaliada segundo a função objetivo $f(x)$ é apresentada na Figura 4.



Figura 4 – Representação de uma solução para o problema com as variáveis de decisão na forma binária.

Na terminologia dos AGs cada indivíduo ou cromossomo representa uma solução para o problema que se propõem resolver. Fazendo uma analogia entre os indivíduos ou cromossomos com o esquema de solução para o problema, cada *gene* equiivale à uma alternativa de manejo (X_{ij}). Aos possíveis estados assumidos por cada *gene*, dá-se o nome de *alelo*. À posição ou endereço dos *genes* no cromossomo denomina-se *locus*. Cada cromossomo é avaliado segundo uma função de aptidão ou *fitness*, usualmente igual à função objetivo ($f(x)$) do problema (Figura 5)

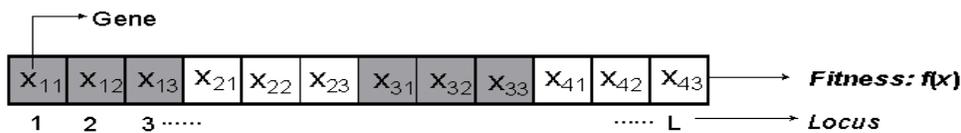


Figura 5 – Cromossomo com seus respectivos *genes*, *locus* e *fitness*.

A codificação binária foi utilizada para representar os indivíduos, e uma particularidade na criação de cromossomo, utilizada neste trabalho, foi a de assinalar os valores dos genes de forma que os cromossomos gerados representassem apenas soluções viáveis, segundo a restrição de singularidade,

ou seja, ao conjunto de genes j ($j = 1, 2, \dots, N$) que representam as variáveis de decisão em cada unidade de manejo i ($i = 1, 2, \dots, M$), um único gene recebe valor igual a um ($X_{ij} = 1$) e os demais genes recebem valores iguais a zero ($X_{ij} = 0$)

(Figura 6).



Figura 6– Representação de um cromossomo com seus alelos na forma binária.

2.3.4. Função de avaliação

A função de avaliação do AG, denominada de função de aptidão ou *fitness*, é utilizada para medir o desempenho dos indivíduos, podendo ser determinada de diversas formas. A forma mais comum em problemas de otimização é a utilização da própria função objetivo do problema. Então, nos modelos utilizados para representar os problemas de planejamento florestal, a aptidão ou *fitness* de cada indivíduo pode ser obtida por:

$$f(x) = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} \quad (5)$$

em que:

$f(x)$ = é a função objetivo utilizada para representar a aptidão ou *fitness* dos indivíduos;

X_{ij} = variável de decisão (gene) que representa a j -ésima alternativa de manejo adotada na i -ésima unidade de manejo;

C_{ij} = é o valor presente líquido total (\$) de cada unidade de manejo i , colhida segundo a alternativa j ;

A expressão (5) é útil para avaliar aqueles indivíduos que representam soluções viáveis para o problema. Entretanto, a geração de indivíduos com estas características pode ser um processo muito oneroso devido a dificuldades, em muitos problemas, de se identificar as suas soluções viáveis. O teste da viabilidade dos indivíduos (soluções) pode ser dispensado, ao se utilizar funções de avaliação apropriadas. Uma das formas alternativas, é permitir que os indivíduos inviáveis permaneçam na população, porém, aplicando-se a eles uma penalidade. Este

artifício é muito utilizado pelos algoritmos genéticos reportados na literatura e foi também implementado neste trabalho, conforme descrito a seguir.

2.3.4.1. Método das penalidades

Considerando que as restrições de singularidade são atendidas no processo de geração dos cromossomos (são gerados somente cromossomos viáveis segundo esta restrição), um dos artifícios para viabilizar o método das penalidades é relaxar as demais restrições do problema (demanda mínima e máxima), alterando o cálculo da *fitness* da seguinte forma:

$$f_p(x) = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} - vp \times VT \quad (6)$$

$$VT = \sum_{k=0}^{H-1} (D_K + S_K) = \sum_{k=0}^{H-1} D_K + \sum_{k=0}^{H-1} S_K = DT + ST \quad (7)$$

$$D_K = Dmin_k - \sum_{i=1}^M \sum_{j=1}^N V_{ijk} X_{ij}; \{k = 0, 1, \dots, H - 1\} \quad (8)$$

$$S_K = \sum_{i=1}^M \sum_{j=1}^N V_{ijk} X_{ij} - Dmax_k; \{k = 0, 1, \dots, H - 1\} \quad (9)$$

em que:

$f_p(x)$ = valor da *fitness* (função objetivo) penalizada (\$);

VT = violação total (m^3) das restrições de produção mínima e máxima;

D_K = déficit de produção (m^3) no k -ésimo período do horizonte de planejamento, quando a restrição de produção mínima é violada;

S_K = superávit de produção (m^3) no k -ésimo período do horizonte de planejamento, quando a restrição de produção máxima é violada;

DT = déficit total de produção (m^3);

ST = superávit total de produção (m^3);

vp = penalização (\$/ m^3) para cada unidade de produção violada.

2.3.5. Geração da população inicial

Ao conjunto de indivíduos utilizado para representar um conjunto de soluções para problema em um determinado instante, denomina-se população. O AG inicia-se com uma população inicial (P_0) e através de repetidas operações criam novas populações (P_n) até que um critério de parada seja atendido. A população inicial é, portanto, um ponto de partida para o AG. Existem algumas formas para criação dessa população, e dentre elas a geração aleatória tem sido muito utilizada pelos AGs descritos na literatura. Neste trabalho, optou-se, também, pela geração aleatória da população inicial. Para tal, escolheu-se aleatoriamente uma variável de decisão (gene) em cada unidade de manejo para receber o valor 1 (um) e, assinalou-se para as demais variáveis daquela unidade, o valor 0 (zero), satisfazendo a restrição de singularidade. Por exemplo, para o problema 1, considerando uma população de tamanho (n) igual a cinco ($n=5$), uma configuração válida dos cromossomos (indivíduos) para representar esta população é mostrada na (Figura 7).

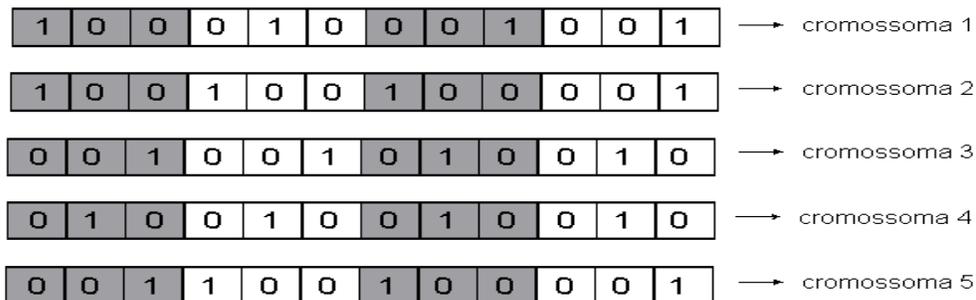


Figura 7 - Geração aleatória de uma população inicial com cinco indivíduos.

2.3.6. Seleção dos indivíduos

A seleção consiste em escolher os indivíduos da população que irão se reproduzir (cromossomos pais). Existem vários métodos de seleção descritos na literatura. Neste trabalho, foram testados cinco métodos de seleção distintos: proporcional, escalonamento, *Boltzmann*, ranqueamento e *tournament*.

No método de seleção proporcional, cada indivíduo i de uma população de tamanho n possui uma expectativa de sobrevivência no instante t (neste trabalho equivale a iteração corrente do algoritmo) associada à função de avaliação (aptidão ou a *fitness*), dada por:

$$E_{it} = \frac{f_p(x)_{it}}{\sum_{i=1}^n f_p(x)_{it}} \times 100 \quad (10)$$

em que:

E_{it} = expectativa de sobrevivência do i -ésimo indivíduo no instante t ;

$f_p(x)_{it}$ é a função de avaliação ou *fitness* do indivíduo i no instante t .

A seleção por escalonamento é um segundo método de seleção utilizado pelo AG. Este método utiliza a função *sigma scaling* para calcular a expectativa de sobrevivência dos indivíduos, dada por:

$$E_{it} = \begin{cases} 1 + \frac{f_p(x)_{it} - \bar{f}_p(x)_t}{2\mathbf{S}(t)}, & \text{se } \mathbf{S}(t) \neq 0 \\ 1 & \text{se } \mathbf{S}(t) = 0 \end{cases} \quad (11)$$

em que:

$\bar{f}_p(x)_t$ = *fitness* média da população no instante t ;

$\mathbf{S}(t)$ = desvio padrão da *fitness* da população no instante; e demais variáveis conforme já definidas.

Um outro método utilizado pelo AG implementado neste trabalho é a seleção de *Boltzmann*. Neste método a expectativa de sobrevivência dos indivíduos é calculada utilizando a seguinte expressão:

$$E_{it} = \frac{e^{f_p(x)_{it}/T}}{E_t} \quad (12)$$

$$E_t = \sum_{i=1}^n e^{f_p(x)_{it}/T} / n \quad (13)$$

em que:

T = temperatura;

E_t = expectativa de sobrevivência média da população no tempo t .

A temperatura (T) inicia com um valor alto, sendo gradativamente reduzida até a convergência do algoritmo. Vários programas de redução da temperatura

podem ser adotados, e nenhum trabalho até o momento apresenta sugestões de qual o melhor esquema de redução dessa temperatura. Dois parâmetros são utilizados para definir a temperatura (T) em um dado instante: a temperatura inicial (T_0), a taxa de redução da temperatura (R_T). Neste trabalho, o esquema utilizado para fornecer a temperatura em um determinado instante foi a função potencial dada por:

$$T_k = T_0 \left(1 - \frac{R_T}{100} \right)^k \quad (14)$$

em que:

T_k = temperatura na k -ésima iteração;

T_0 = temperatura inicial;

R_T = taxa de redução da temperatura, expressa em percentagem;

k = iteração.

O método de ranqueamento utilizado neste trabalho foi o mesmo proposto por Baker e descrito por MITCHELL (1996). O procedimento de seleção por este método é efetuado da seguinte forma: após calculado a *fitness* penalizado de cada indivíduo da população no instante t (k -ésima iteração), os indivíduos são classificados (ranqueados) em ordem crescente de *fitness* penalizado (de 1 até n). Em seguida, a expectativa de sobrevivência de cada indivíduo i na população na t -ésima iteração é calculada por:

$$E_{it} = Min + (Max - Min) \frac{rank(it) - 1}{N - 1} \quad (15)$$

Os valores de *Min* e *Max* correspondem aos valores mínimos e máximos, do maior e menor rank, respectivamente. De acordo com MITCHELL (1996), o valor ideal dos parâmetros *Max* e *Min* são: $1 \leq Max \leq 2$ e $Min = 2 - Max$.

Na seleção *tournament*, também denominada por seleção por jogos, dois cromossomos pais são escolhidos ao acaso na população para se reproduzirem. Em seguida, escolhe-se um parâmetro q e um número aleatório r uniformemente distribuído entre 0 e 1. Se $r < q$, os dois indivíduos são selecionados para serem pais; caso contrário os dois indivíduos são recolocados na população original e

podem ser selecionados novamente. O valor de q utilizado neste trabalho foi igual a 0,75.

2.3.7. Implementação dos métodos de seleção

Com exceção do método de *tournament*, onde os pares de indivíduos selecionados aleatoriamente são cruzados entre si, os demais métodos de seleção propostos neste trabalho necessitam de uma operação adicional para viabilizar a seleção dos indivíduos que sofrerão cruzamento. Tendo calculado a expectativa de sobrevivência dos indivíduos através de um método de seleção, um mecanismo probabilístico deve ser utilizado para escolha dos melhores indivíduos, levando em consideração as suas expectativas de sobrevivência. O método da *roleta russa* é um dos principais métodos utilizados neste processo, com referências na maioria dos trabalhos de AG propostos na literatura.

O método da roleta russa implementado neste trabalho, consiste em associar a cada indivíduo uma fatia da roleta igual a sua probabilidade de sobrevivência (P_i) dada por:

$$P_i = \frac{E_{it}}{\sum_{i=1}^n E_{it}} \quad (16)$$

De acordo com a expressão acima, a probabilidade de sobrevivência do indivíduo é proporcional à sua expectativa de sobrevivência. A implementação computacional deste método foi feita da seguinte forma. Após calculada a probabilidade de sobrevivência (P_i) de cada indivíduo, atribuiu-se a cada indivíduo uma fatia (F_i) na roleta igual à sua probabilidade de sobrevivência P_i . Os indivíduos são assinalados sequencialmente na roleta, conforme esquematizado no exemplo da Figura 6, onde cada indivíduo i da população tem a sua fatia (F_i) na roleta compreendida no intervalo $L_{i-1} \leq F_i < L_i$ (sendo, $L_0 = 0$ e $L_i = L_{i-1} + P_i$). Por exemplo, o intervalo da roleta para o indivíduo 1 é $(0 \leq F_1 < 0.2)$, para o indivíduo 2 é $(0.2 \leq F_2 < 0.35)$ e assim, até o último indivíduo (5) com o intervalo dado por $(0.87 \leq F_5 \leq 1)$. Após definidos os intervalos que cada indivíduo ocupa na roleta, um número

aleatório r ($0 \leq r \leq 1$) é gerado para simular uma rodada da roleta. O indivíduo selecionado em cada rodada da roleta é aquele cujo número aleatório gerado pertence ao intervalo do respectivo indivíduo na roleta. Por exemplo, de acordo com o esquema da roleta da Figura 8. se um número aleatório $r = 0,15$ fosse sorteado, o indivíduo 1 seria escolhido. O processo é repetido até que a nova população seja restabelecida.

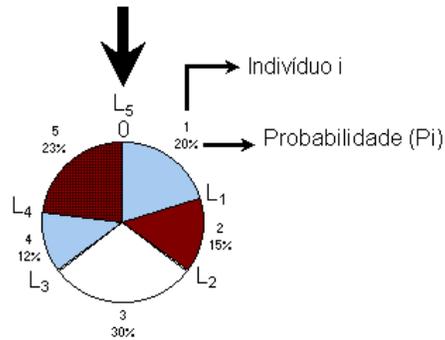


Figura 8 - Esquema de uma roleta russa para uma população (n) de cinco indivíduos.

2.3.8. Operadores genéticos

Os operadores genéticos (\mathbf{y}) são mecanismos que garantem a evolução dos indivíduos, criando, a partir de uma dada população inicial (P_0) novas populações (P_i) ou gerações (G_i) de indivíduos melhorados, ($P_0 \xrightarrow{\mathbf{y}} P_1 \xrightarrow{\mathbf{y}} P_2 \dots \xrightarrow{\mathbf{y}} P_t \equiv G_0 \xrightarrow{\mathbf{y}} G_1 \xrightarrow{\mathbf{y}} G_2 \dots \xrightarrow{\mathbf{y}} G_t$). A cada nova população ou geração se espera melhoria no desempenho dos indivíduos em relação ao meio ambiente, que traduzido para o processo de busca de novas soluções pelo AG, espera-se uma melhoria na qualidade das novas soluções geradas.

Os operadores genéticos utilizados neste trabalho foram o *crossover* e a mutação, conforme descritos a seguir.

2.3.8.1. Operador de *crossover*

Este operador permite a obtenção de novos indivíduos (filhos), a partir da combinação (cruzamento) dos cromossomos dos pais. A operação de *crossover* pode

ser resumida da seguinte forma: utilizando um método de seleção qualquer, dois indivíduos *pais* da população são selecionados e cruzados entre si para gerar dois novos indivíduos (*filhos*) para a nova geração. A probabilidade de cruzamento dos dois indivíduos pais é conhecida como taxa de *crossover* (T_C). O cruzamento dos dois indivíduos pais é efetuado toda vez que um número aleatório r , com distribuição uniforme, selecionado entre o intervalo 0 e 1 for menor do que a taxa de *crossover* ($r \leq T_C$). Este processo é efetuado até que o tamanho da população seja restabelecido.

Os tipos de *crossover* mais utilizados nos AGs são de um único ponto, de dois pontos e multipontos. O AG desenvolvido neste trabalho utiliza estes três tipos de *crossover* e são detalhados a seguir.

2.3.8.1.1. *Crossover* de um único ponto

No *crossover* de um único ponto os dois indivíduos pais selecionados para reproduzir têm seus materiais genéticos (correspondem aos estados ou valores dos genes) trocados a partir de um ponto de ruptura, escolhido aleatoriamente (Figura 9). Para preservar a estrutura original do cromossomo, evitando a geração de cromossomos filhos que representem apenas soluções viáveis segundo as restrições de singularidade, foi utilizado um *crossover* especial. Neste *crossover*, restringe-se os pontos de quebra em um conjunto de pontos pertencentes ao grupo de posições que represente as interfaces das unidades de manejo (Figura 9), o que garante a geração de novos indivíduos com a características desejadas.

Apesar dos filhos gerados serem indivíduos que representam soluções viáveis segundo a restrição de singularidade, o mesmo pode não ocorrer com as demais restrições do problema (produção mínima e máxima, por exemplo). Ainda assim, seria possível forçar o AG a trabalhar com populações constituídas somente por indivíduos que representassem apenas soluções viáveis para o problema. A complexidade inerente dos espaços de soluções da maioria dos problemas dificulta a formação de populações constituídas apenas por indivíduos com estas características, podendo elevar o custo computacional do procedimento ou mesmo

inviabilizar a corrida do algoritmo devido à dificuldade em encontrar somente indivíduos que representem soluções viáveis. Um procedimento alternativo é permitir a formação de populações constituídas por indivíduos que representem tanto soluções viáveis quanto soluções inviáveis para o problema, aplicando, porém, penalidades àqueles indivíduos que violarem alguma restrição do problema. Este segundo procedimento foi o escolhido para a implementação do AG proposto.

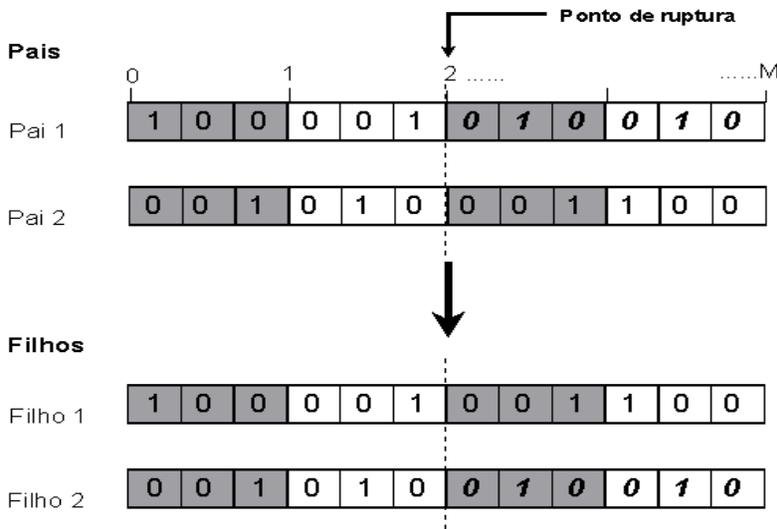


Figura 9– *Crossover* de um ponto gerando novos indivíduos.

2.3.8.1.2. *Crossover* de dois pontos

Neste tipo de *crossover*, após selecionado os dois cromossomos pais que sofrerão *crossover*, dois pontos de ruptura são escolhidos aleatoriamente e o material genético dos respectivos cromossomos entre os dois pontos escolhidos são trocados entre si. O esquema da Figura 10 representa um *crossover* de dois pontos escolhidos ao acaso.

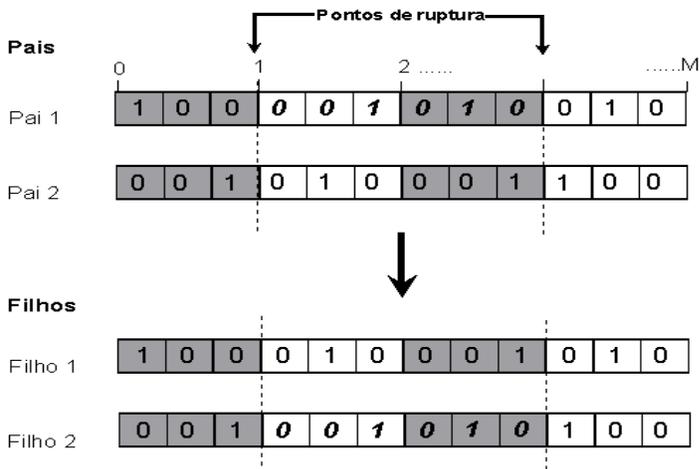


Figura 10 – *Crossover* de dois pontos gerando novos indivíduos.

2.3.8.1.3. *Crossover* multipontos

O *crossover* multipontos foi implementado neste trabalho da seguinte forma: após o par de cromossomos pais ter sido selecionado para sofrer *crossover*, três ou mais pontos de ruptura são escolhidos ao acaso e, o material genético entre eles são trocados de acordo com uma regra pré-definida. A regra utilizada neste trabalho é a seguinte: os pontos de ruptura escolhidos ao acaso são ordenados em ordem crescente para evitar sobreposição dos intervalos de troca. Iniciando com o primeiro ponto selecionado na escala de ordenamento, a troca do material genético entre os dois cromossomos pais é efetuada toda vez que o ponto de ruptura imediatamente à direita do ponto escolhido for um número par. Caso contrário, não se efetua a troca dos materiais genéticos e cada filho herda exatamente as mesmas características do seu respectivo pai, ou seja, o filho 1 recebe a carga genética do pai 1 naquele intervalo e, da mesma forma, o filho 2 recebe a carga genética do pai 2 no respectivo intervalo. O procedimento é repetido até que todos os pontos de ruptura sejam percorridos. A Figura 11 representa um esquema de *crossover* com quatro pontos de ruptura (0, 2, 3 e $M = 4$). Neste exemplo, existem 3 pares (intervalos) possíveis de trocas (0-2, 2-3 e 3-4). De acordo com a regra de troca a permutação dos materiais genéticos só ocorrerá entre os pares de pontos 0-2 e 3-4, uma vez que o vizinho à direita dos pontos 0 e 3 são, respectivamente, 2 e 4. A troca entre o par 2-3

não ocorreu por motivo inverso, ou seja, o vizinho à direita do ponto de ruptura 2 é um número ímpar (3).

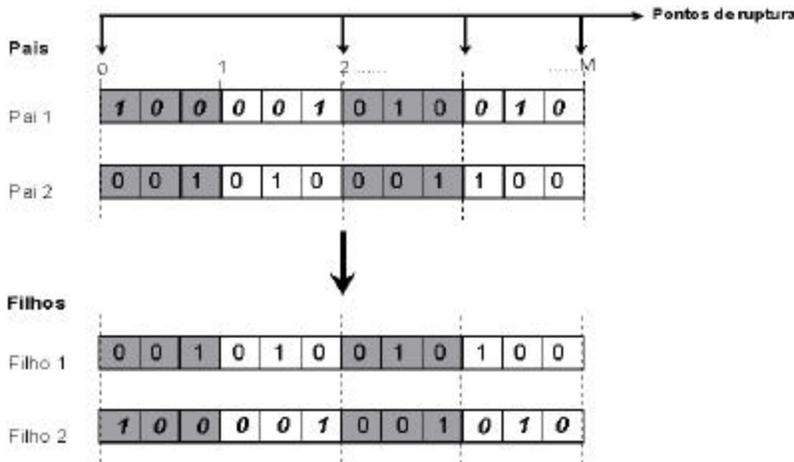


Figura 11 – *Crossover* multiponto gerando novos indivíduos.

2.2.8.2. Operador de mutação

O operador de mutação altera o valor de alguns bits (alelos) do cromossomo, conferindo diversidade à população. As posições de um ou mais genes são escolhidas ao acaso e, os valores dos seus *alelos* invertidos. A mutação é, em geral, aplicada a todos os cromossomos da população, ou seja, em 100% dos indivíduos. Para o AG desenvolvido neste trabalho foi definido um novo parâmetro para permitir a ocorrência da mutação com taxas inferiores a 100% dos indivíduos. Este parâmetro foi denominado como *taxa de cromossomo mutados* (T_{CM}) e refere-se à percentagem dos cromossomos da população que sofrerão mutação numa dada corrida do AG. A *probabilidade de mutação* (P_m), também conhecida como *taxa de mutação* (T_m), por outro lado, determina a percentagem de *genes* mutados em cada cromossomo. A Figura 12 mostra um esquema de mutação pontual, onde após escolhido o cromossomo 1 e os seus genes a serem mutados, os valores de seus alelos foram invertidos de 1 para 0 e vice-versa.

O operador de mutação pode, entretanto, alterar a condição de viabilidade dos cromossomos. A Figura 12 ilustra dois problemas com a operação de mutação. No primeiro caso, após mutação do cromossomo 1, todos os genes (i.e. as variáveis

de decisão) da unidade de manejo 2 (cromossomo 2) possuem valores nulos. No segundo caso, ocorre quando mais de um gene é assinalado com valor igual a 1 em uma mesma unidade de manejo, conforme pode ser visto na unidade de manejo 3 do cromossomo 2. Assim, ao se desejar o assinalamento de exatamente uma alternativa de manejo (gene) em cada unidade de manejo e evitar a geração de cromossomos inviáveis, segundo a restrição de singularidade, torna-se necessário corrigir o problema. Tal correção é feita através de uma operação pós-mutação. Esta operação foi efetuada da seguinte forma: na unidade de manejo em que todas as variáveis possuem valor nulo, uma nova variável, diferente daquela escolhida para ser mutada, é escolhida aleatoriamente para receber valor 1, garantindo o assinalamento de exatamente uma variável em cada unidade de manejo. No caso da violação da restrição de singularidade, para corrigir o problema, altera-se o valor da variável, com valor igual a 1 antes da mutação, para 0 (zero). O cromossomo três da Figura 12 ilustra a operação pós-mutação para corrigir estes problemas.

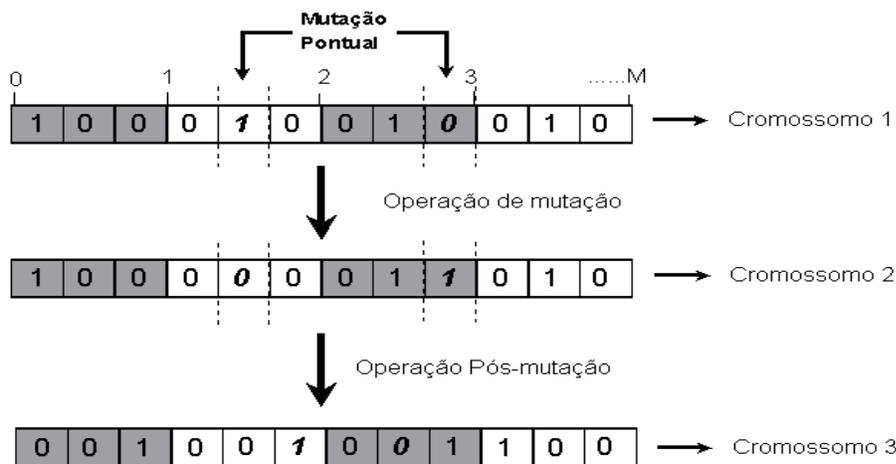


Figura 12 - Operação de mutação e pós-mutação para obtenção de cromossomos viáveis.

2.3.9. Critério de parada

Os procedimentos iterativos requerem a definição de um critério de parada para interromper a corrida do algoritmo. Segundo YOUSSEF et al. (2001) um dos problemas enfrentados na confecção do AG é determinar o critério de parada. Esta dificuldade deve-se à inabilidade ou dificuldade desses algoritmos em avaliar a

qualidade da solução num dado instante da busca, pois, em geral são míopes (i.e. geralmente não há informação de quanto a solução corrente se aproxima da solução ótima). Há, contudo, alguns mecanismos para definição do critério de parada. O número máximo de iterações, o tempo máximo de processamento, a estabilização da função objetivo, são alguns exemplos. A escolha desse critério pode também afetar a qualidade da solução encontrada. Por exemplo, ao se utilizar como critério de parada, a primeira solução viável encontrada, uma solução de baixa qualidade pode ser escolhida, uma vez que para um dado problema pode existir uma infinidade de soluções viáveis. O AG proposto utiliza como critério de parada um mecanismo mais inteligente e é baseado no seguinte procedimento: após encontrar a primeira solução viável (i.e. com violação total igual a zero, $VT=0$), o AG executa um número de interações adicionais (Δk) para tentar melhorar a solução encontrada. Portanto, o AG tenta encontrar várias soluções viáveis, escolhendo aquela de melhor valor para a função objetivo. Para interromper a corrida do AG após um número de iterações sem encontrar uma solução viável para o problema, foi definido um número máximo de iterações (k_{max}).

2.3.10. Escolha dos parâmetros do AG

Um AG possui, em geral, duas categorias de parâmetros: qualitativos e quantitativos. Os principais parâmetros qualitativos são os tipo de *crossover* (um ponto, dois pontos e múltiplos pontos) e os métodos de seleção (proporcional, escalonamento, *Boltzmann*, ranqueamento e *tournament*). Os principais parâmetros quantitativos são constituídos pelo tamanho da população (n), taxa de *crossover* (T_c) e taxa de mutação (T_m). Além dos parâmetros básicos, o AG desenvolvido neste trabalho inclui outros parâmetros utilizados com diferentes finalidades, tais como: o valor da penalização (vp), utilizado nos cálculos das penalidades; o número de interações adicionais (Δk), para melhorar a qualidade da solução utilizada no critério de parada; o número máximo de iterações permitidas (k_{max}); a temperatura inicial (T_0) e a taxa de redução de temperatura (T_R), utilizados pelo método de

seleção de *Boltzmann* e o valor máximo para o maior rank (*Max*) utilizados pelo método de seleção por ranqueamento.

A Figura 13 mostra uma janela da rotina computacional, com opções para escolha dos parâmetros utilizados em uma dada corrida do AG.

Existem muitos trabalhos desenvolvidos com a finalidade de dimensionar os valores dos parâmetros na literatura de computação evolucionária, em muitas áreas da ciência. Entretanto, não há resultados conclusivos sobre os melhores valores desses parâmetros para todas as aplicações. Muitas aplicações usam valores reportados em outros trabalhos, outras definem os seus próprios valores com base em experimentação. Para os problemas de planejamento florestal propostos, não existe sequer trabalhos reportando a utilização dos AGs para solucioná-los, o que sugere o uso da experimentação para dimensioná-los. Os parâmetros e seus respectivos níveis ou valores testados nesse trabalho são mostrados no Quadro 3. As combinações entre os níveis e parâmetros apresentados neste Quadro gera um ensaio com 60 diferentes combinações. Cada combinação pode ser vista como um cenário ou tratamento diferente. O objetivo é escolher o melhor cenário ou tratamento. Os valores dos parâmetros T_C , T_m , T_o , T_R e Max foram escolhidos baseados no trabalho de GARCIA et al. (2000). Os demais valores de parâmetros foram escolhidos após um ensaio piloto fornecendo algumas diretrizes para sua escolha.

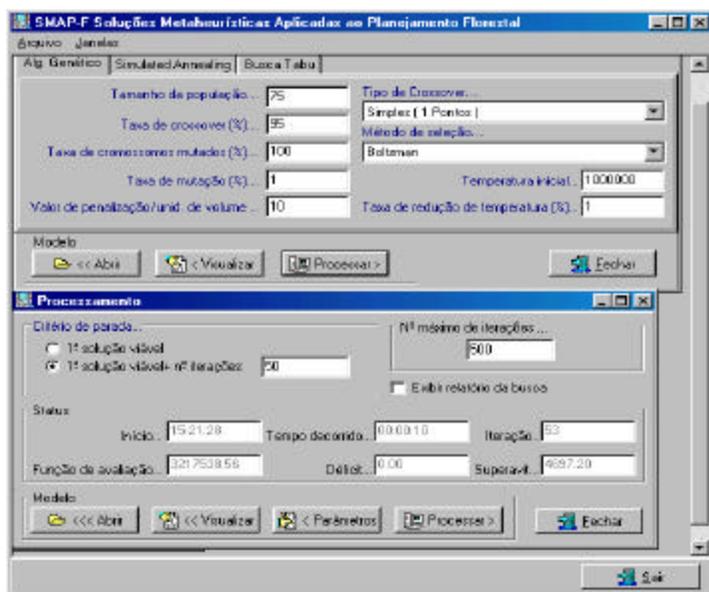


Figura 13 – SMAP-F: interface computacional para definição dos parâmetros do AG.

Para comparar os métodos de seleção e tamanho da população utilizou-se apenas o *crossover* de um único ponto e os demais parâmetros fixados, conforme sugerido por GARCIA et al. (2000). Este ensaio foi feito para os problemas 2, 3, 4 e 5 utilizando-se 4 repetições para cada tratamento, gerando um total de 320 observações.

Os tipos de *crossover* foram comparados utilizando-se, além dos parâmetros sugeridos por GARCIA et al. (2000), o melhor método de seleção e tamanho da população selecionados com base no ensaio prévio deste trabalho. Este ensaio foi feito também nos problemas 2, 3, 4 e 5, utilizando-se 8 repetições por tratamento, o que gerou um total de 96 observações.

Quadro 3 - Resumo dos parâmetros utilizados pelo algoritmo genético

Parâmetro	Níveis	Valores ou categorias dos parâmetros
-----------	--------	--------------------------------------

Método de seleção	5	I	II	III	IV	V
Tamanho da população (n)	4	25	50	75	100	
Tipo de <i>crossover</i>	3	1P	2P	MP		
Taxa de <i>crossover</i> (T_C) - %	1	95				
Taxa de cromossomo mutados (T_{CM}) - %	1	100				
Taxa de mutação (T_m) - %	1	1.0				
Valor da penalização (vp) – \$/m ³	1	10				
Temperatura inicial (T_0)	1	10 ⁶				
Taxa de redução de temperatura (T_R) - %	1	1				
Valor para rank máximo (Max)	1	1.1				
Número máximo de iteração (k_{max})	1	500				
Interações adicionais (Δk)	1	50				

em que: I = seleção proporcional; II = seleção por escalonamento; III = seleção de *Boltzmann*; IV = seleção ranqueamento; V = seleção *tournament*; 1P = *crossover* de um ponto; 2P = *crossover* de dois pontos; e MP = *crossover* multipontos.

2.3.11. Avaliação do desempenho do AG

A performance do AG desenvolvido neste trabalho foi avaliada através da sua eficácia e eficiência. A eficácia foi calculada pela razão, expressa em porcentagem, entre o melhor valor da função objetivo (*fitness*) do AG e o ótimo matemático, obtido pelo algoritmo exato *branch and bound*, assim:

$$Ef = \frac{f_{AG}}{f_o} \times 100 \quad (17)$$

em que:

Ef = eficácia (%);

f_{AG} = valor da melhor solução obtida em uma corrida do AG (\$);

f_o = valor da solução ótima obtida pelo algoritmo *branch and bound* (\$);

As medidas adequadas de eficiência dependem de estudos mais sofisticados, como a análise de complexidade de algoritmos, tais como complexidade de tempo e de espaço. Neste trabalho, eficiência do AG foi avaliada utilizando-se apenas o tempo de processamento do algoritmo, o que permite apenas comparações relativas entre os métodos de solução, processadas utilizando-se os mesmos recursos computacionais.

2.3.12. Análise estatística dos resultados

As comparações entre os métodos de seleção, tamanho da população e tipo de *crossover* foram efetuadas utilizando-se o procedimento estatístico proposto por LEITE e OLIVEIRA (2000) para comparação de métodos analíticos. O procedimento proposto por estes autores, denominado teste L&O, utiliza uma metodologia que combina o teste F de GRAYBILL ($F(H_0)$), o teste t para o erro médio ($t_{\bar{e}}$) e o coeficiente de correlação linear ($r_{Y_j Y_1}$) entre as variáveis. Na Tabela 1 é apresentado um resumo da regra decisória deste teste. Além do teste L&O utilizou-se a estatística descritiva para auxiliar a análise dos resultados.

Tabela 1 - Regra de decisão do teste de L&O para comparações de metodos analíticos.

Situação	$F(H_0)$	$t_{\bar{e}}$	$r_{Y_j Y_1}$	Decisão
1	ns	ns	$r_{Y_j Y_1} \geq (1 - \bar{e})$	$Y_j = Y_1$
2	ns	ns	$r_{Y_j Y_1} \leq (1 - \bar{e})$	$Y_j \neq Y_1$
3	ns	*	$r_{Y_j Y_1} \geq (1 - \bar{e})$	$Y_j \neq Y_1$
4	ns	*	$r_{Y_j Y_1} \leq (1 - \bar{e})$	$Y_j \neq Y_1$
5	*	ns	$r_{Y_j Y_1} \geq (1 - \bar{e})$	$Y_j \neq Y_1$
6	*	ns	$r_{Y_j Y_1} \leq (1 - \bar{e})$	$Y_j \neq Y_1$
7	*	*	$r_{Y_j Y_1} \geq (1 - \bar{e})$	$Y_j \neq Y_1$
8	*	*	$r_{Y_j Y_1} \leq (1 - \bar{e})$	$Y_j \neq Y_1$

ns e * denotam, respectivamente, não significativo e significativo para o nível α de probabilidade

3. RESULTADOS E DISCUSSÃO

3.1. O plano de manejo

A rotina computacional desenvolvida para implementar o AG apresenta um relatório com o melhor cromossomo (indivíduo), representando uma solução para o problema, com seus respectivos genes (variáveis de decisão) e *fitness* (valor da função objetivo). Além, disso, os valores dos níveis das restrições (para o problema em foco, a produção anual) é apresentado, conforme mostrado na Figura 14.

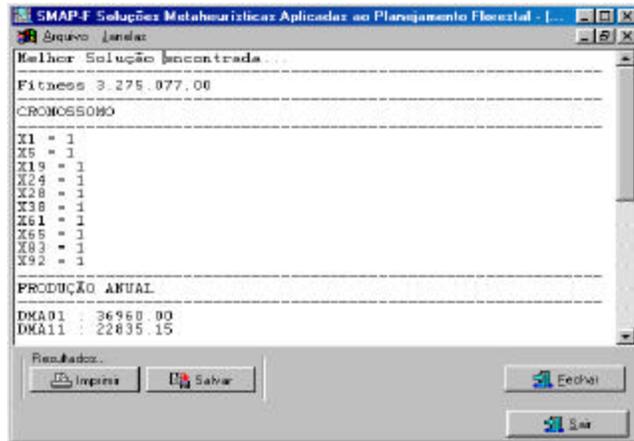


Figura 14 – *SMAP-F*: esquema de um relatório com o melhor cromossomo, *fitness* e níveis das restrições para o problema.

A comparação de uma solução obtida via AG e Programação Inteira (PI) para o problema 2 é apresentada no Quadro 4. Nesta simulação, o valor da *fitness* do AG corresponde a 97,72% do valor ótimo, obtido através do algoritmo de PI *branch and bound*. Apenas uma alternativa de manejo (X_{ij}) é alocada em cada unidade de manejo, o que garante o atendimento da restrição de singularidade, pelas duas abordagens de solução. As unidades de manejo que tiveram diferentes alocações de alternativas de manejo entre as duas abordagens de solução foram as unidades 3, 4, 5 e 6.

Quadro 4 - Solução via AG e PI para o problema 2 com as unidades de manejo e suas respectivas alternativas de manejo

Unidade de manejo	AG: <i>fitness</i> (\$) = 3.275.077,00		PI: função objetivo (\$) = 3.351.434,00	
	Variável	Valor	Variável	Valor
1	X_4	1	X_4	1
2	X_8	1	X_8	1
3	X_{20}	1	X_{22}	1
4	X_{24}	1	X_{26}	1
5	X_{29}	1	X_{35}	1
6	X_{38}	1	X_{41}	1
7	X_{64}	1	X_{64}	1
8	X_{73}	1	X_{73}	1
9	X_{89}	1	X_{89}	1
10	X_{91}	1	X_{91}	1

O fluxo de produção de madeira apresentado na Figura 15 mostra que a demanda mínima de 15.000,00 m³ e máxima de 60.000,00 m³ estabelecidas foram atendidas pelas duas abordagens de solução. Apenas no período oito, a solução via Programação Inteira (PI) forneceu uma produção superior ao Algoritmo Genético (AG).

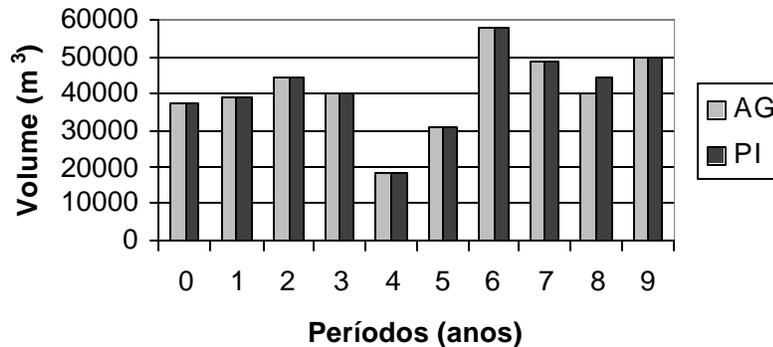


Figura 15 – Fluxo de madeira para o plano de manejo resultante da solução por AG e PI.

3.2. Ilustração de uma corrida do AG

Para ilustrar uma corrida do AG, utilizou-se o problema 1 descrito no item 2.1 deste capítulo. O modelo para representar este problema contém 12 variáveis de decisão, quatro restrições de singularidade e quatro restrições de produção mínima. Uma população inicial com 16 indivíduos (cromossomos) gerados aleatoriamente é apresentada no Quadro 5. De acordo com este quadro, todos os indivíduos gerados violam as restrições de produção mínima, uma vez que a violação total é não nula ($VT > 0$). Muitas aplicações de AG propostos na literatura efetuam o teste de viabilidade das soluções geradas, ou seja, testam o cromossomo segundo todas as restrições do problema e incluem na lista de indivíduos da população apenas cromossomos viáveis. Esta estratégia apresenta-se como principal desvantagem, o alto custo computacional para testar a viabilidade dos indivíduos. A estratégia utilizada no AG proposto neste trabalho permite que a população apresente em sua composição, indivíduos viáveis ($VT = 0$) e inviáveis ($VT > 0$). Esta estratégia, além

de dispensar o teste de viabilidade das restrições, diminuindo o custo computacional, aumenta a diversidade da população, ao permitir que os indivíduos inviáveis se reproduzam. A violação de restrição do problema proposto é avaliada pela variável violação total (VT). Ressalta-se ainda que não há necessidade de testar a viabilidade das restrições de singularidade, já que o algoritmo proposto gera somente indivíduos viáveis, segundo esta restrição. É evidente também que se existissem restrições de outras naturezas no problema tais como: produção máxima, adjacência, reforma etc, elas deveriam ser também testadas.

Cada indivíduo foi avaliado segundo sua *fitness* penalizada $f_p(x)$. Considerando, por exemplo, os indivíduos 10 e 16 representados no Quadro 5, ambos com a *fitness* não penalizada de \$ 338.000,00, os mesmos indivíduos, quando avaliados segundo a função a *fitness* penalizada, têm suas novas aptidões dadas por \$ 238.000,00 e \$ 288.000,00, respectivamente. Portanto, o indivíduo 16 é mais apto do que o indivíduo 10 e terá mais chances de se reproduzir, segundo este critério. Pode-se notar pelo mesmo Quadro 5 que a maior aptidão do indivíduo 16 é propiciada pela menor violação total que ele proporciona na restrição de produção mínima, sendo de 10.000,00 m³ contra os 20.000, 00 m³ do indivíduo 10. Um outro exemplo, interessante para avaliar o efeito da função *fitness* na classificação da aptidão dos indivíduos pode ser visto ao se comparar os indivíduos 5 e 16. Ao se utilizar a função objetivo sem penalização, o indivíduo 5 apresenta uma maior aptidão que o indivíduo 16, uma vez que sua *fitness* é igual a \$ 343.000,00, contra os \$338.000,00 do indivíduo 16. Entretanto, ao se utilizar a função objetivo penalizada, o indivíduo 16 (*fitness* penalizada igual 288.000,00) passa a apresentar uma maior aptidão que o indivíduo 5 (*fitness* penalizada 243.000,00).

Uma vez gerada a população inicial e calculada a *fitness* de cada indivíduo, utilizando o método de seleção proporcional, foram selecionados os indivíduos para reprodução, ou seja, os cromossomos pais que serão cruzados entre si para gerarem novos indivíduos (cromossomos filhos). A probabilidade (P_i) de cada indivíduo reproduzir é apresentada no Quadro 5. De posse destas probabilidades e utilizando o método da *roleta russa* (Figura 16), os indivíduos são selecionado aleatoriamente. Para o exemplo em foco, a roleta foi girada 16 vezes (sendo 16, o tamanho da

população), onde em cada rodada um indivíduo foi selecionado. Cada indivíduo possui uma faixa assinalada na roleta igual a sua probabilidade (P_i).

Quadro 5 – População inicial e suas respectivas medidas de aptidão, violação total, probabilidade de sobrevivência e classe de probabilidade²

IND	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	$f(x)$	$fp(x)$	VT	P_i	Li	Ls	
1	1	0	0	0	1	0	1	0	0	0	0	1	366000	316000	10000	7,2	0,0	7,2	
2	1	0	0	1	0	0	0	0	1	1	0	0	298000	238000	12000	5,4	7,2	12,7	
3	0	0	1	0	0	1	1	0	0	0	1	0	378000	328000	10000	7,5	12,7	20,1	
4	0	1	0	1	0	0	0	0	1	1	0	0	318000	218000	20000	5,0	20,1	25,1	
5	0	1	0	1	0	0	0	1	0	0	1	0	343000	243000	20000	5,6	25,1	30,7	
6	0	0	1	0	0	1	0	1	0	0	0	1	433000	383000	10000	8,7	30,7	39,4	
7	1	0	0	1	0	0	1	0	0	1	0	0	268000	168000	20000	3,8	39,4	43,3	
8	0	1	0	0	0	1	0	1	0	1	0	0	333000	233000	20000	5,3	43,3	48,6	
9	1	0	0	0	1	0	1	0	0	0	1	0	326000	226000	20000	5,2	48,6	53,7	
10	0	0	1	1	0	0	0	0	1	1	0	0	338000	238000	20000	5,4	53,7	59,2	
11	0	1	0	0	1	0	0	1	0	0	1	0	361000	261000	20000	6,0	59,2	65,1	
12	0	0	1	0	0	1	0	0	1	0	0	1	448000	348000	20000	7,9	65,1	73,1	
13	1	0	0	1	0	0	0	1	0	0	0	1	363000	303000	12000	6,9	73,1	80,0	
14	0	1	0	0	0	1	1	0	0	0	0	1	398000	346000	10400	7,9	80,0	87,9	
15	0	0	1	0	1	0	0	1	0	1	0	0	341000	241000	20000	5,5	87,9	93,4	
16	1	0	0	0	0	1	1	0	0	0	1	0	338000	288000	10000	6,6	93,4	100	
													Média	273625	Total	100,0			

² Li e Ls são, respectivamente, o limite inferior e superior da classe de probabilidade dos indivíduos.

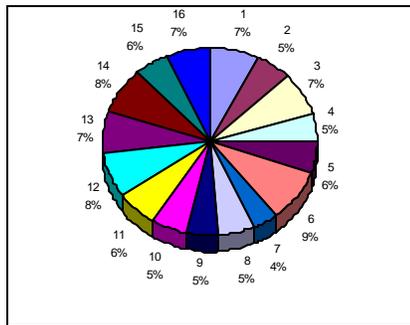


Figura 16 - Esquema da seleção proporcional através do método da roleta russa.

Os indivíduos selecionados para se reproduzirem estão apresentados no Quadro 6. Estes indivíduos, conforme já ressaltado, representam os cromossomos pais que irão se cruzar para produzir os cromossomos filhos da próxima geração. Por exemplo, os dois primeiros pais são representados pelos indivíduos 14 e 11. Observe que oito pares de cromossomos pais são gerados pelo processo de seleção,

produzindo 16 novos indivíduos (cromossomos filhos). Uma vez selecionado os pares de cromossomos para cruzamento ou *crossover*, gerou-se aleatoriamente um número aleatório r_c ($0 < r_c < 1$) que foi comparado com a probabilidade ou taxa de *crossover* ($P_C = 0,95$). Todos os pares de cromossomos que tiveram $r_c < P_C$ sofreram *crossover*. Portanto, somente um par de cromossomo (15 e 14) não sofreu cruzamento, uma vez que r_c é maior do que 0,95. Isto significa que os indivíduos 15 e 14 serão passados para a próxima geração sem nenhuma alteração genética através da operação de *crossover*.

Após selecionado os pares de cromossomos para cruzamento, escolheu-se então a posição no cromossomo para efetuar o *crossover*. Note que as posições onde se permite a quebra dos cromossomos é aquela representada pelas interfaces dos genes entre duas unidades de manejo, evitando que os novos cromossomos gerados violem a restrição de singularidade. As posições de quebra foram também geradas aleatoriamente. No *crossover* de um único ponto o material genético entre os dois cromossomos é trocado a partir do ponto de quebra. Para exemplificar, observe os pares de cromossomos pais 14 e 11. Após escolhido a quebra dos cromossomos na posição 1 o material genético entre os cromossomos pais são trocado a partir desta posição. Os cromossomos filhos gerados após o *crossover* estão apresentados no Quadro 6.

Quadro 6 - Cromossomos pais a serem cruzados, selecionados através do método da seleção proporcional

		Posição de quebra para <i>crossover</i>												
		1			2			3						
Posição de <i>Crossover</i>	r_c	Pais	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}
1	0,55	14	0	1	0	0	0	1	1	0	0	0	0	1
		11	0	1	0	0	1	0	0	1	0	0	1	0
3	0,61	10	0	0	1	1	0	0	0	0	1	1	0	0
		3	0	0	1	0	0	1	1	0	0	0	1	0
2	0,88	12	0	0	1	0	0	1	0	0	1	0	0	1
		1	0	1	0	1	0	0	0	0	1	1	0	0
1	0,56	4	0	1	0	1	0	0	0	0	1	1	0	0
		2	1	0	0	1	0	0	0	0	1	1	0	0
3	0,78	5	0	1	0	1	0	0	0	1	0	0	1	0
		3	0	0	1	0	0	1	1	0	0	0	1	0

1	0,97	15	0	0	1	0	1	0	0	1	0	1	0	0
		14	0	1	0	0	0	1	1	0	0	0	0	1
3	0,88	12	0	0	1	0	0	1	0	0	1	0	0	1
		2	1	0	0	1	0	0	0	0	1	1	0	0
1	0,34	7	1	0	0	1	0	0	1	0	0	1	0	0
		15	0	0	1	0	1	0	0	1	0	1	0	0

A mutação é um outro fenômeno que pode ocorrer nos indivíduos da população, com uma dada probabilidade (taxa de mutação P_m), geralmente menor do que 0,01. Para o exemplo em foco, cada gene apresenta uma probabilidade de mutação dada por 1/12, ou seja, de 0,083. Este maior valor de P_m foi adotado para viabilizar a ilustração desse operador, pois ao se adotar uma probabilidade de mutação menor ou igual a 0,01, nenhum gene seria mutado neste exemplo. Na mutação pontual, um gene é escolhido aleatoriamente para ter o seu alelo invertido (Quadro 7). Por exemplo, o cromossomo (filho) 3 teve o gene 5 escolhido aleatoriamente para sofrer a mutação.

Quadro 7- Mutação pontual dos cromossomos filhos após o cruzamento dos cromossomos pais

Posição de mutação	Filhos	Posição para efetuar mutação											
		1	2	3	4	5	6	7	8	9	10	11	12
		X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}
-	1	0	1	0	0	1	0	0	1	0	0	1	0
-	2	0	1	0	0	0	1	1	0	0	0	0	1
5	3	0	0	1	1	0	0	0	0	1	0	1	0
-	4	0	0	1	0	0	1	1	0	0	1	0	0
11	5	0	0	1	0	0	1	0	0	1	1	0	0
-	6	0	1	0	1	0	0	0	0	1	0	0	1
9	7	0	1	0	1	0	0	0	0	1	1	0	0
-	8	1	0	0	1	0	0	0	0	1	1	0	0
-	9	0	1	0	1	0	0	0	1	0	0	1	0
4	10	0	0	1	0	0	1	1	0	0	0	1	0
-	11	0	0	1	0	1	0	0	1	0	1	0	0
6	12	0	1	0	0	0	1	1	0	0	0	0	1
-	13	0	0	1	0	0	1	0	0	1	0	0	1
-	14	1	0	0	1	0	0	0	0	1	1	0	0
3	15	1	0	0	0	1	0	0	1	0	1	0	0

9	16	0	0	1	1	0	0	1	0	0	1	0	0
---	----	---	---	---	---	---	---	---	---	---	---	---	---

No Quadro 8 são apresentados os cromossomos filhos após a mutação ter sido efetuada em alguns indivíduos. Os alelos destacados tiveram seus valores invertidos de 0 para 1 ou vice-versa. Por exemplo, no cromossomo 3 o gene (variável de decisão) 5 com alelo igual 0 foi alterado para 1, enquanto o cromossomo 6 teve o alelo do gene 9 alterado de 1 para 0. A ocorrência da mutação altera as condições de viabilidade dos indivíduos, alterando inclusive, a restrição de singularidade. Para restabelecer a viabilidade nas restrições de singularidade, uma operação pós-mutação é efetuada. O cromossomo 3, por exemplo, possui após mutação dois genes em uma mesma unidade de manejo com valores iguais a 1, o que viola a restrição de adotar um único regime em cada unidade de manejo (Quadro 8). Neste caso, para restabelecer a condição de viabilidade do cromossomo 3, basta alterar o valor do gene 4, o qual tinha valor igual a 1 antes da mutação, para 0 (Quadro 9).

Pode ocorrer também que todos os genes de uma mesma unidade tenham seus valores iguais a 0, o que também representa uma violação da restrição de adotar exatamente um único regime em cada unidade de manejo. Uma destas situações é representada pelo cromossomo 6, após a mutação ter alterado o alelo do gene 9 de 1 para 0 (Quadro 8). Para corrigir este problema escolheu-se aleatoriamente um novo gene na respectiva unidade de manejo para ter o valor de seu alelo igual a 1 (Quadro 9).

Quadro 8 – Situação dos cromossomos após a mutação ter sido efetuada

Posição de mutação	Filhos	Posição para efetuar mutação											
		1	2	3	4	5	6	7	8	9	10	11	12
		X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}
-	1	0	1	0	0	1	0	0	1	0	0	1	0
-	2	0	1	0	0	0	1	1	0	0	0	0	1
5	3	0	0	1	1	1	0	0	0	1	0	1	0
-	4	0	0	1	0	0	1	1	0	0	1	0	0
11	5	0	0	1	0	0	1	0	0	1	1	1	0
-	6	0	1	0	1	0	0	0	0	0	0	0	1
-	7	0	1	0	1	0	0	0	0	1	1	0	0

-	8	1	0	0	1	0	0	0	0	1	1	0	0
-	9	0	1	0	1	0	0	0	1	0	0	1	0
4	10	0	0	1	1	0	1	1	0	0	0	1	0
-	11	0	0	1	0	1	0	0	1	0	1	0	0
6	12	0	1	0	0	0	0	1	0	0	0	0	1
-	13	0	0	1	0	0	1	0	0	1	0	0	1
-	14	1	0	0	1	0	0	0	0	1	1	0	0
3	15	1	0	1	0	1	0	0	1	0	1	0	0
9	16	0	0	1	1	0	0	1	0	1	1	0	0

A nova geração de indivíduos (cromossomos filhos) obtida pelo cruzamento e mutação é apresentada no Quadro 9. Uma vez que o critério de parada, neste exemplo considerado como a primeira solução viável, foi atendido ($VT=0$), encerra-se o procedimento de busca, com o melhor indivíduo representado pelo cromossomo 4. Se o critério de parada não tivesse sido alcançado, o algoritmo continuaria a busca, repetindo as operações de seleção, *crossover* e mutação até que ele fosse atendido. Pode-se verificar também que o AG propiciou uma geração de indivíduos mais evoluídos, uma vez que o *fitness* médio da população inicial que era de \$ 273.625,00 (Quadro 5, anterior) passou para \$ 299.733,00 (Quadro 9), um aumento médio de 2,7%.

O melhor cromossomo encontrado pelo AG representa também a solução ótima para o problema quando resolvido pelo algoritmo exato *branch and bound*. Vale ressaltar, entretanto, que para problemas maiores e, ou mais complexos a solução ótima dificilmente será obtida. O que se obtêm, na maioria das vezes são uma ou mais soluções boas.

Quadro 9 - Nova geração de indivíduos após *crossover* e mutação terem sido efetuados

IN	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	$f(x)$	$fp(x)$	VT	Pi	Li	Ls
D																		
1	0	1	0	0	1	0	0	1	0	0	1	0	361000	261000	20000	5,8	0,0	5,8
2	0	1	0	0	0	1	1	0	0	0	0	1	398000	346000	10400	7,7	5,8	13,5
3	0	0	1	0	1	0	0	0	1	0	1	0	396000	246000	30000	5,5	13,5	19,0
4	0	0	1	0	0	1	1	0	0	1	0	0	338000	338000	0	7,5	19,0	26,5
5	0	0	1	0	0	1	0	0	1	0	1	0	408000	308000	20000	6,9	26,5	33,3
6	0	1	0	1	0	0	1	0	0	0	0	1	368000	318000	10000	7,1	33,3	40,4

7	0	1	0	1	0	0	0	0	1	1	0	0	318000	218000	20000	4,8	40,4	45,3
8	1	0	0	1	0	0	0	0	1	1	0	0	298000	238000	12000	5,3	45,3	50,6
9	0	1	0	1	0	0	0	1	0	0	1	0	343000	243000	20000	5,4	50,6	56,0
10	0	0	1	1	0	0	1	0	0	0	1	0	348000	290000	11600	6,5	56,0	62,4
11	0	0	1	0	1	0	0	1	0	1	0	0	341000	241000	20000	5,4	62,4	67,8
12	0	1	0	0	1	0	1	0	0	0	0	1	386000	384000	400	8,5	67,8	76,3
13	0	0	1	0	0	1	0	0	1	0	0	1	448000	348000	20000	7,7	76,3	84,1
14	1	0	0	1	0	0	0	0	1	1	0	0	298000	238000	12000	5,3	84,1	89,3
15	0	0	1	0	1	0	0	1	0	1	0	0	341000	241000	20000	5,4	89,3	94,7
16	0	0	1	1	0	0	0	0	1	1	0	0	338000	238000	20000	5,3	94,7	100,0
Média													299733	Total	100,0			

3.3. Avaliação dos parâmetros e medidas de performance do AG

Os resultados do teste L&O para comparar os métodos de seleção, tamanho da população e tipo de *crossover* são mostrados no Quadro 10. Com exceção do tipo de *crossover*, houve diferenças estatísticas significativas entre os níveis ou categorias dos parâmetros para os problemas analisados a um nível de significância igual a 1%.

Apenas os métodos de seleção por escalonamento e de *Boltzmann* não apresentaram diferenças estatísticas significativas entre si pelo teste de L&O a um nível de 1% de probabilidade (Quadro 10). Na Figura 17a é apresentado as medidas de eficácia dos cinco métodos de seleção avaliados. Pode-se observar que os métodos de seleção de *Boltzmann* e escalonamento apresentaram resultados superiores aos demais métodos. Esta superioridade é reforçada pelas Figuras 17b, uma vez que os métodos de seleção de *Boltzmann* e escalonamento, quando comparados aos demais, apresentaram uma maior frequência de valores mais altos de eficácia. A eficácia foi medida pela relação entre o valor da *fitness* do AG e o ótimo matemático obtido através da solução do problema, utilizando-se o algoritmo exato *branch and bound*. De acordo com as Figuras 17a e, o pior método de seleção foi o de *tournament*.

A tendência de resultados melhores apresentados pelo método de seleção por escalonamento e *Boltzmann* já eram esperados, uma vez que estes métodos possuem mecanismos de seleção teoricamente mais eficientes que os demais. Os métodos de seleção de *Boltzmann* e escalonamento utilizam mecanismos diferenciados de

pressão de seleção dos indivíduos ao longo da busca utilizando-se de informações do problema. Enquanto a seleção por escalonamento mantém uma pressão de seleção aproximadamente constante ao longo da busca, utilizando-se informações do desvio padrão da população, a seleção de *Boltzmann* altera a pressão de seleção ao longo da busca utilizando um mecanismo de resfriamento ou diminuição da temperatura (MITCHELL, 1996). A variação da pressão de seleção pelo método de *Boltzmann* é conseguida através da diminuição da temperatura. Com a temperatura maior no início da busca, a expectativa de sobrevivência dos indivíduos tornam-se próxima de um, o que propicia uma probabilidade igual de sobrevivência. Com a corrida do algoritmo, a pressão de seleção vai sendo aumentada através da redução da temperatura com os indivíduos mais aptos tendo maiores chances de sobrevivência. Ao contrário da seleção por escalonamento e *Boltzmann*, a seleção *tournament* ou por jogos é totalmente aleatória, não utilizando nenhuma informação do problema para seleção dos indivíduos ao longo da busca. Os piores resultados desse método ressaltam as vantagens dos métodos mais informados.

Vale ressaltar que cuidados especiais devem ser tomados na utilização dos métodos de seleção de *Boltzmann* e ranqueamento, uma vez que tais métodos utilizam parâmetros adicionais que podem afetar a performance do AG. Considerando que os parâmetros desses métodos foram fixados baseados em experiências de outros problemas é possível que tais escolhas não sejam as mais adequadas para uma boa performance do AG resolver os problemas em foco. A exemplo de outros parâmetros, estes poderiam ter sido ajustados para os problemas propostos através de experimentação prévia. O método de seleção de *Boltzmann*, além de utilizar parâmetros adicionais, pode utilizar diferentes programas de redução de temperatura e não foram avaliados neste trabalho.

Outro aspecto que merece ser destacado, refere-se aos critérios de parada adotados. Recordando que o critério de parada adotado utilizou um número de iterações adicionais (Δk), para melhorar a primeira solução viável encontrada, percebe-se que a escolha deste valor pode representar um aspecto muito importante na qualidade da solução encontrada pelo AG. Uma escolha balanceada de Δk é decisiva para uma boa performance do AG. Valores muito baixos podem diminuir a

eficácia do algoritmo devido à redução do número de soluções viáveis encontradas nesse menor intervalo, e desta forma, diminuindo as chances de encontrar solução de melhor qualidade entre as soluções candidatas. Valores muito altos, apesar de aumentar as chances de encontrar uma solução de melhor qualidade, podem depreciar a eficiência do AG, causando depreciação no tempo de processamento da solução. A dificuldade em escolher um valor de Δk adequado deve-se a uma possível relação desse parâmetro com o tamanho e complexidade dos problemas, tamanho de população, dentre outros aspectos do AG. Considerando que neste trabalho este valor foi escolhido subjetivamente, é possível que o valor adotado não tenha sido o mais adequado e, por consequência, depreciado a eficácia e a eficiência do AG. Existem diversos outros critérios de parada e necessitam ser melhores investigados.

Os resultados do teste L&O para o tamanho de população mostraram que apenas os tamanhos iguais a 75 e 100 não diferiram entre si a um nível de probabilidade de 1%. Nas Figuras 17c e 17d são mostrados, respectivamente, a eficácia e as suas respectivas frequências para os diferentes tamanhos de população estudados. De acordo com estas figuras os tamanhos de população iguais a 75 e 100 propiciaram melhores eficácias para o AG. Segundo YOUSSEF et al. (2001), uma das dificuldades que surgem na confecção de um AG é definir o tamanho de população. Segundo estes autores, uma população pequena pode causar convergência prematura do algoritmo, enquanto uma população muito grande pode demandar muito tempo de processamento para o algoritmo encontrar uma solução boa.

Não houve diferenças estatísticas significativas do tipo de *crossover* na eficácia do AG, de acordo com o teste de L&O a um nível de 1% de probabilidade (Quadro 10).

Pode-se observar ainda, na Figura 17, que a eficácia do AG oscilou entre os valores de 80 e 100% para os diferentes métodos de seleção e tamanho da população. Entretanto, ao escolher os melhores métodos de seleção (*Boltzmann* e escalonamento) e tamanho de população (75 e 100), o AG apresentou eficácia média

de 94,28%, valor mínimo igual a 90,01%, valor máximo igual a 98,48% com coeficiente de variação de 2,08%.

Apesar da eficiência do AG não ter sido avaliada rigorosamente nesse trabalho, o que requereria análise de complexidade de algoritmos mais sofisticadas tais como complexidade de tempo e de espaço, os resultados computacionais preliminares já mostraram vantagens do AG em relação ao algoritmo exato *branch and bound*. O tempo computacional gasto pelo algoritmo *branch and bound* cresce mais rapidamente com o tamanho do problema (número de variáveis), comparado ao AG. Para o problema 4 descrito no item 2.1 deste capítulo, com 423 variáveis de decisão, o algoritmo *branch and bound* consumiu um tempo cinco vezes superior ao AG (Figura 18).

Quadro 10 - Teste de L&O³ para comparar os efeitos dos parâmetros da metaheurística Algoritmo Genético

Métodos de seleção		F(H0)	t _e	r _{yij}
Proporcional	Escalonamento	ns	*	sim
Proporcional	<i>Boltzmann</i>	ns	*	sim
Proporcional	Rankeamento	ns	*	sim
Proporcional	<i>Tournament</i>	ns	*	sim
Escalonamento	<i>Boltzmann</i>	ns	ns	sim
Escalonamento	Rankeamento	ns	*	sim
Escalonamento	<i>Tournament</i>	ns	*	sim
<i>Boltzmann</i>	Rankeamento	ns	*	sim
<i>Boltzmann</i>	<i>Tournament</i>	ns	*	sim
Rankeamento	<i>Tournament</i>	ns	*	sim
Tamanho da população		F(H0)	t _e	r _{yij}
25	50	ns	ns	sim
25	75	ns	*	sim
25	100	ns	*	sim
50	75	ns	*	sim
50	100	ns	*	sim
75	100	ns	ns	sim
Tipo de <i>crossover</i>		F(H0)	t _e	r _{yij}
Um ponto	Dois pontos	ns	ns	sim
Um ponto	Multipontos	ns	ns	sim
Dois pontos	Multipontos	ns	ns	sim

* = significativo a 1%; ns = não significativo e sim = correlação positiva

³ Regra decisória descrita no item 2.3.12. deste capítulo

As vantagens em termos de tempo de processamento do AG em relação aos métodos exatos de solução podem ser melhorados através de aperfeiçoamentos na estrutura do algoritmo. Além disso, é necessário aumentar a capacidade do AG para resolver problemas maiores e mais complexos, a fim de que possam ser utilizados para solução de problemas reais de planejamento florestal. Para aumentar a velocidade de processamento do AG é necessário considerar a possibilidade de adotar uma estrutura de dados para otimizar operações como geração das populações e suas respectivas avaliações, como cálculo da *fitness* e testes dos indivíduos, segundo as restrições do problema (avaliação de sua viabilidade). É consenso na literatura especializada que tais operações são críticas e consomem a maioria dos recursos de processamento do AG (MITCHELL, 1996).

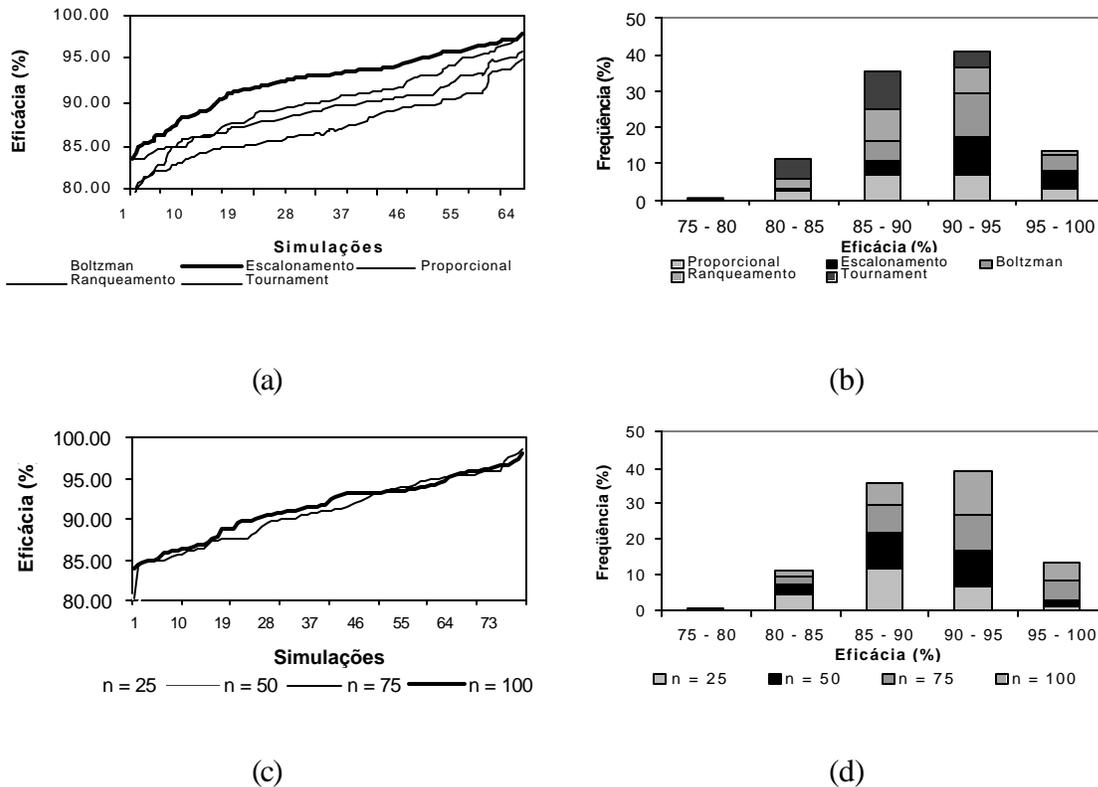


Figura 17 - Medida de eficácia e respectivas frequências para diferentes métodos de seleção (a e b) e tamanho da população (c e d) do algoritmo genético.

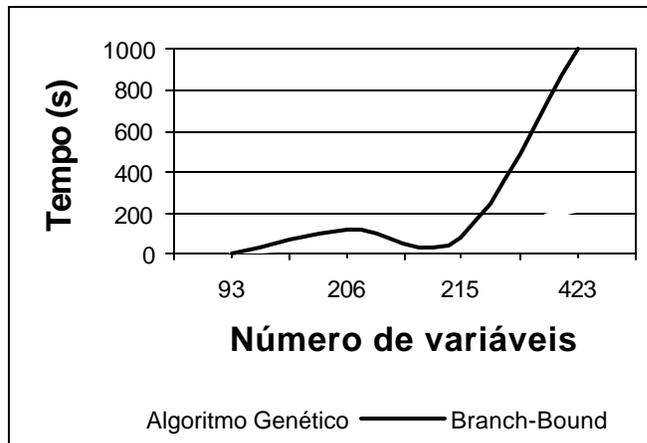


Figura 18 – Tempo de solução obtida através do AG e o algoritmo *branch and bound*.

4. CONCLUSÕES

Este trabalho teve como objetivo principal desenvolver um Algoritmo Genético (AG) para resolver problemas de planejamento florestal de natureza combinatorial. A estrutura dos problemas estudados consistiram da maximização da receita líquida, sujeita às restrições de singularidade, produção mínima e máxima. Utilizando a rotina computacional *SMAP-F* (Soluções Metaheurísticas Aplicadas ao Planejamento Florestal), foram testados os efeitos de diferentes parâmetros como método de seleção, tamanho da população e tipo de *crossover*. Houve diferenças estatísticas significativas a um nível de 1% de probabilidade entre os níveis ou categorias de alguns parâmetros avaliados. Os melhores métodos de seleção foram os de *Boltzmann* e escalonamento, e os melhores tamanhos de população foram iguais a 75 e 100 indivíduos. O tipo de *crossover* não afetou a eficácia do AG. Considerando a igualdade estatística dos resultados, o *crossover* de um ponto parece mais atrativo devido à sua simplicidade e facilidade de implementação.

Os melhores parâmetros do AG forneceram, em média, soluções que alcançaram 94,28% do valor ótimo, chegando a atingir valores máximos de 98,48%

do ótimo, com um tempo de processamento cinco vezes inferior ao principal algoritmo exato de solução (*branch and bound*). Os resultados mostraram também que os métodos de seleção mais informados (*Boltzmann* e escalonamento) foram os que produziram melhores resultados. Dentre os métodos de seleção estudados, o método de *Boltzmann* necessita ser melhor avaliado, uma vez que neste estudo os valores de seus parâmetros foram fixados baseados em experiências de outros trabalhos. Ainda, em relação ao método de *Boltzmann*, outros esquemas de redução da temperatura devem ser avaliados.

As vantagens do AG desenvolvido em relação a algumas técnicas heurísticas convencionais descritas na literatura para resolver problemas de planejamento florestal é a flexibilidade em comportar restrições de diversas naturezas sem, contudo, alterar a estrutura do algoritmo. Ao contrário, em muitas heurísticas convencionais, a inclusão de novas restrições requer alterações drásticas em suas estruturas ou é até mesmo impossível de implementar devido às limitações de seus modelos em tratar problemas específicos.

Aplicabilidade universal dos AGs propiciada pela flexibilidade desses algoritmos em resolver problemas de natureza variada, permite concluir que estes podem ser utilizados como alternativa atraente para resolver outros problemas do setor florestal. Alguns exemplos, são aqueles envolvendo restrições espaciais, como as restrições de colheita em talhões adjacentes para evitar a abertura de extensas clareiras, roteamento de veículos para o transporte florestal e diversos outros problemas de natureza combinatorial encontrados no gerenciamento florestal, seja ele em florestas equiâneas ou inequiâneas.

Com base neste estudo pôde-se concluir que os AGs foram adequados para solução de importantes problemas de planejamento florestal, de difíceis soluções pelos algoritmos exatos da atualidade.

Os objetivos desse trabalho foram alcançados. Entretanto, melhorias na capacidade da rotina computacional desenvolvida necessita ser implementada para viabilizar a sua aplicação em problemas maiores;

A melhor performance do Algoritmo Genético é dependente da escolha de uma boa configuração de seus parâmetros. Como não existe conclusões seguras

para a maioria dos problemas de planejamento florestal, em especial, os problemas maiores e, ou mais complexos, essa configuração deve ser escolhida baseada na experimentação.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- GARCIA, A.L.; GARCIA, B.; FRIEDMANN, C.V. Resolução de um problema de equilíbrio de trabalho e distribuição de agentes de serviço. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000. p923-935.
- GOLDBERG, D.E. **Genetic algorithms in search, optimization and machine learning**. Addison Wesley. 1989.
- HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**, Ann Harbor, University of Michigan Press, 1975.
- JOHNSON, K. N., SCHEURMAN, H. L. Techniques for prescribing optimal timber harvest and investment under different objectives - discussion and synthesis. **Forest Science**, Washington, 18(1), p.1-31, 1977.
- LEITE, H.G. & OLIVEIRA, F.H.T. Statistical Procedure to Test the Identity of Analytical Methods. **Comm. Soil Sci. Plant Anal.**, 2001, 16p. (no prelo)
- MITCHELL, M.** An introduction to genetic algorithms. **London, A Bradford Book, 1996. 205p.**
- REBELLO, F.R.; HUMACHER, S. Uma proposta de algoritmo genético de duas fases para roteamento de veículos. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. **Anais**. Viçosa: UFV, 2000.
- YOUSSEF, H.; SAIT, S.M.; ADICHE, H. Evolutionary algorithms, simulated annealing and tabu search: a comparative study. **Engineering Applications of Artificial Intelligence**. 14: 167-181, 2001.

CAPÍTULO 5

METAHEURÍSTICA BUSCA TABU PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL COM RESTRIÇÕES DE INTEIREZA

RESUMO – Este trabalho teve como objetivo desenvolver e testar um algoritmo baseado na metaheurística Busca Tabu (BT), para a solução de problemas de gerenciamento florestal com restrições de inteireza. Os problemas avaliados continham entre 12 e 423 variáveis de decisão, sujeitos às restrições de singularidade, produção mínima e produção máxima periódicas. Todos os problemas tiveram como objetivo a maximização do valor presente líquido. O algoritmo para implementação da BT foi codificado em ambiente *delphi* 5.0 e os testes efetuados em um microcomputador AMD K6II 500 MHZ, com memória RAM de 64 MB e disco rígido de 15GB. A performance da BT foi avaliada de acordo com as medidas de eficácia e eficiência. Os diferentes valores ou categorias dos parâmetros da BT foram testados e comparados quanto aos seus efeitos na eficácia do algoritmo. A seleção da melhor configuração de parâmetros foi feita utilizando o teste L&O, a um nível de 1% de probabilidade e análises através de estatísticas descritivas. A melhor configuração de parâmetros propiciou à BT uma eficácia média de 95,97%, valor mínimo igual a 90,39%, valor máximo igual a 98,84%, com um coeficiente de variação de 2,48% do ótimo matemático. Para o problema de maior porte a eficiência da BT foi duas vezes superior à eficiência do algoritmo exato *branch and bound*, apresentando-se como uma abordagem muito atrativa para solução de importantes problemas de gerenciamento florestal.

Palavras-chave: Gerenciamento Florestal, heurísticas, Busca Tabu.

METAHEURISTIC TABU SEARCH TO SOLVER FOREST PLANNING PROBLEM WITH INTEGER CONSTRAINTS

ABSTRACT – This work was objective to develop and test an algorithm based in Tabu Search (TS) metaheuristic to solve problems of forest management with integer constraints. TS was tested in five problems containing between 12 and 423 decision variables, subjected to singularity constraints, minimum and maximum production periodically. All the problems had the objective of maximizing the net present value. TS was codified into *delphi* 5.0 language and the tests were performed in a microcomputer AMD K6II 500 MHZ, with RAM memory of 64 MB and hard disk of 15GB. The TS performance was evaluated according to the efficacy and efficiency measures. The different values or categories for the TS parameters were tested and compared in relation to their effects on the algorithm efficacy. The selection of the parameters' best configuration was performed by using the L&O test at 1% probability and analyses via descriptive statistics. The parameters' best configuration provided for TS average efficacy of 95,97%, minimum value equal to 90,39%, maximum value equal to 98,84%, with coefficient of variation being 2,48% of the mathematical optimum, obtained by the exact algorithm *branch and bound*. As for the larger problem, the efficiency of TS was two times superior to the efficiency of the exact algorithm *branch and bound*. TS came out as a quite attractive new approach in forest management for solving important problems of difficult solution or even impossible of being solved through the use of the current computational instruments.

Key-words: Forest Management, Heuristic and Tabu Search.

1. INTRODUÇÃO

Os princípios básicos da Busca Tabu (BT) têm sido usados de diferentes maneiras e com diversos nomes há aproximadamente 20 anos (GLAZAR, 2000). Ela teve início em procedimentos combinatórios em problemas de

cobertura não lineares no fim da década de 70 (GLOVER, 1977). As idéias básicas de sua forma atual foram desenvolvidas por GLOVER (1986) e WERRA e HERTZ (1989) e aperfeiçoada por GLOVER e LAGUNA (1997). A BT é aplicada a diversos tipos de problemas combinatoriais. Os primeiros foram: sequenciamento da produção, balanceamento de canais de comunicação e análise de agrupamentos. Recentemente ela tem sido aplicada ao problema do caixeiro viajante, coloração de grafos, projetos de circuitos integrados, roteamento de veículos, dentre outros (GLAZAR, 2000). Alguns exemplos de problemas de otimização combinatoria no gerenciamento florestal incluem aqueles com restrições de singularidade, adjacência, roteamento de veículos no transporte florestal, corte na indústria de móveis e de papel, dentre outros. Apesar do sucesso da BT na solução de problemas de diferentes áreas, existem poucos trabalhos reportando a sua utilização na solução de problemas de gerenciamento florestal. Alguns exemplos incluem os trabalhos de BOSTON e BETTINGER (1999); LAROZE e GLEBER (1997); LAROZE (1999) e RICHARDS et al. (2000). No Brasil, não existe ainda nenhum trabalho reportando a utilização da BT na solução de problemas de planejamento florestal. Assim, idealizou-se este estudo com os seguintes objetivos:

- desenvolver e testar um algoritmo de Busca Tabu para resolver problemas de planejamento florestal;
- ilustrar aplicação do procedimento em um problema de pequeno porte;
- testar o efeito dos parâmetros da BT na estratégia de solução;
- expandir a aplicação do algoritmo para problema maiores e mais complexos, através do desenvolvimento de um código computacional para este fim.

2. MATERIAL E MÉTODOS

2.1. O Problema de planejamento florestal

Os problemas escolhidos para ilustrar o procedimento da metaheurística Busca Tabu (BT) foram os mesmos propostos para teste do Algoritmo

Genético, apresentados no item 2.1 do capítulo 4, qual seja maximizar o valor presente líquido sujeito às restrições de singularidade (adotar um único regime de manejo em cada talhão), restrições de produção mínima e máxima, periodicamente.

2.2. O Algoritmo BT

O algoritmo BT foi implementado no programa *SMAP-F* (Soluções Metaheurísticas Aplicadas ao Planejamento Florestal), codificado em ambiente *delphi* 5.0. Os testes com o algoritmo foram efetuados em um microcomputador AMD K6II 500 MHZ, com memória RAM de 64 MB e disco rígido de 15GB. O esquema genérico do programa com o respectivo algoritmo foi resumido conforme fluxograma da Figura 1. Os detalhes desse algoritmo são descritos nos tópicos que se seguem.

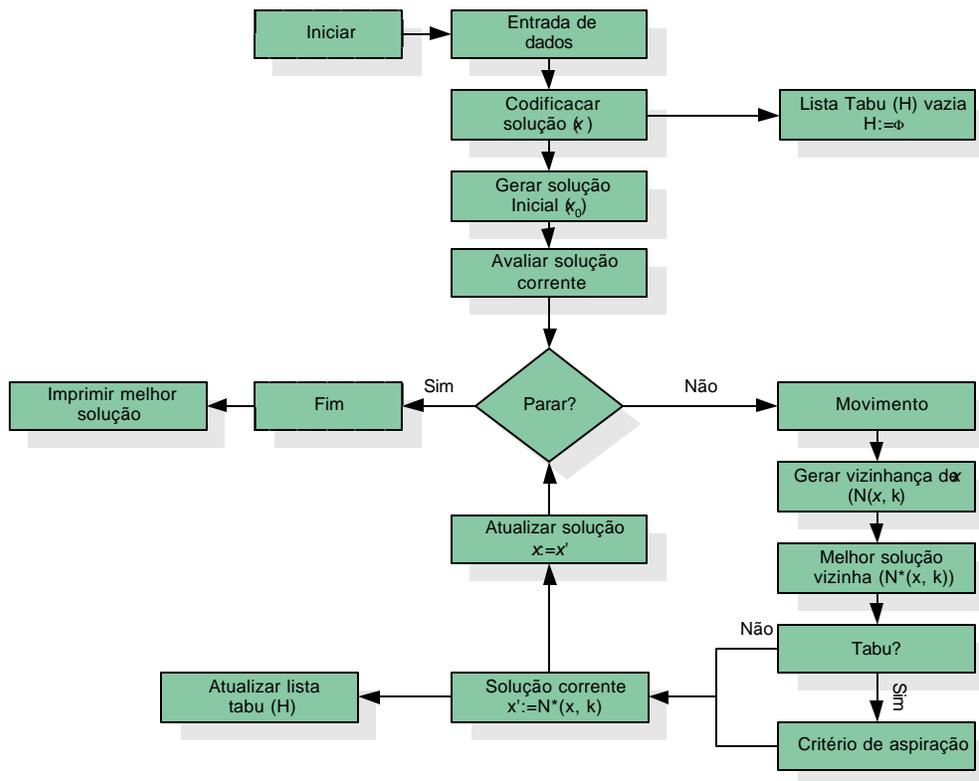


Figura 1 – *SMAP-F*: estrutura geral do programa com respectivo algoritmo para implementação da metaheurística Busca Tabu

2.2.1. Entrada de dados

A exemplo do Algoritmo Genético, a entrada de dados da rotina para implementação da metaheurística BT é um arquivo no formato texto contendo o modelo de Programação Inteira representativo do problema em foco, gerado pelo sistema de planejamento SisFlor. Além do modelo para representar o problema a ser resolvido, a solução de um problema utilizando a metaheurística BT requer a definição dos valores dos seus diferentes parâmetros. Estes valores são definidos pelo usuário em um módulo específico, conforme discutido no item 2.3. deste capítulo.

2.2.2. Codificação da solução

A representação das alternativas de manejo para um dado problema é feita utilizando-se as variáveis de decisão X_{ij} para representar a alternativa de manejo j ($j=1,2,\dots,N$) assinalada à unidade de manejo i ($i =1,2,\dots,M$). Um vetor $V(x) = \{X_{11}, X_{12}, \dots, X_{1j}, X_{21}, X_{22}, \dots, X_{2j}, \dots, \dots, X_{i1}, X_{i2}, \dots, X_{ij}\}$, para representar essas variáveis de decisão foi esquematizado conforme Figura 2.

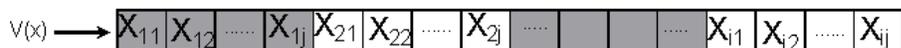


Figura 2 – Representação de um vetor com as variáveis de decisão do problema.

Uma solução (x) viável segundo a restrição de singularidade para o problema, formulada através do modelo de Programação Inteira com as variáveis de decisão na forma binária ($X_{ij} \in \{0,1\}$), avaliada segundo a função objetivo $f(x)$ foi representada de acordo com o esquema da Figura 3.



Figura 3 – Representação de uma solução para o problema com as variáveis de decisão na forma binária.

2.2.3. Solução inicial

A solução inicial (x_0) do algoritmo BT é gerada aleatoriamente. Com uma probabilidade uniformemente distribuída, uma variável de decisão X_{ij} é selecionada aleatoriamente em cada unidade de manejo para receber valor igual a 1 (variável básica). As demais variáveis são assinaladas com valores iguais a 0 (variáveis não básicas), garantindo a viabilidade da restrição de singularidade, ou seja, adotar uma única opção de manejo em cada talhão.

2.2.4. Vizinhança e movimentos

A BT opera sobre a noção de que uma vizinhança $N_i(x,k)$ pode ser construída através de um movimento m para identificar soluções candidatas, onde x é uma solução para o problema e k é a iteração corrente. Estes movimentos são utilizados para fugir de ótimos locais e devem ser hábeis para aceitar um movimento que leve de x para $x' \in N_i(x, k)$, mesmo quando $f(x') < f(x)$, em caso de problemas de maximização.

A vizinhança $N_i(x,k)$ é construída utilizando um mecanismo de movimentos. A estratégia de movimento utilizada neste trabalho explora a vizinhança total ou parcial de uma dada solução x .

2.2.4.1. Vizinhança total

Para gerar a vizinhança total de uma solução x , o mecanismo de movimento empregado, a partir da solução x , efetua todas as possíveis trocas de pares de variáveis (X_{ij}), transformando uma variável não básica ($X_{ij} = 0$) em uma variável básica ($X_{ij} = 1$) e vice-versa. Para problema de otimização discreta com as variáveis $X_{ij} \in \{0,1\}$, a adição (*Adc*) de uma variável na base ($X_{ij} = 1$) implica na remoção (*Rem*) de uma outra variável da base ($X_{ij} = 0$). A título de exemplo, utilizando-se o problema um, com quatro unidades de manejo e 12 variáveis de decisão, uma vizinhança de x é representada no Quadro 1. A cada k iterações são geradas $N_i(x, k)$ soluções vizinhas de x , onde $N_i(x, k)$ é a vizinhança completa de x ,

obtida pelo mecanismo de movimento determinístico de adição e remoção de pares de variáveis em cada unidade de manejo (Quadro 1). A Vizinhança total (V_t) de uma dada solução x pode ser calculada por:

$$V_t = \sum_{i=1}^M (N_i - 1) \quad (1)$$

em que:

V_t = vizinhança total;

M = número de unidades de manejo do problema;

N_i = número de alternativas de manejo em cada unidade de manejo.

Quadro 1 – Vizinhança total de x gerada através do mecanismo de adicionar e remover deterministicamente pares de variáveis X_{ij} em cada unidade de manejo i .

	UM (i)	1			2			3			4			Movimentos	
	Variáveis	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	<i>Adc</i>	<i>Rem</i>
x		0	1	0	1	0	0	0	0	1	1	0	0	-	-
Vizinhança – $N_i(x, k)$	$N_1(x, k)$	1	0	0	1	0	0	0	0	1	1	0	0	X_{11}	X_{12}
	$N_2(x, k)$	0	0	1	1	0	0	0	0	1	1	0	0	X_{13}	X_{12}
	$N_3(x, k)$	0	1	0	0	1	0	0	0	1	1	0	0	X_{22}	X_{21}
	$N_4(x, k)$	0	1	0	0	0	1	0	0	1	1	0	0	X_{23}	X_{21}
	$N_5(x, k)$	0	1	0	1	0	0	1	0	0	1	0	0	X_{31}	X_{33}
	$N_6(x, k)$	0	1	0	1	0	0	0	1	0	1	0	0	X_{32}	X_{33}
	$N_7(x, k)$	0	1	0	1	0	0	0	1	0	0	1	0	X_{42}	X_{41}
	$N_8(x, k)$	0	1	0	1	0	0	0	1	0	0	0	1	X_{43}	X_{41}

O mecanismo, apresentado anteriormente, possui vantagem de, partindo-se de uma solução x é possível gerar a sua vizinhança total, podendo constituir-se numa O mecanismo, apresentado anteriormente, possui vantagem de, partindo-se de uma solução x é possível gerar a sua vizinhança total, podendo constituir-se numa estratégia de busca muito eficiente já que toda vizinhança de x é avaliada. Entretanto, para problemas maiores, o número de soluções vizinhas pode ser muito elevado, depreciando o tempo computacional para convergência do algoritmo. Por exemplo, utilizando a expressão (1) acima, em um problema com apenas 20

unidades de manejo, cada uma contendo 11 alternativas de manejo seriam necessários gerar e avaliar 200 soluções vizinhas. Uma alternativa para diminuir o número de soluções vizinhas é restringir o tamanho da vizinhança. Neste caso apenas uma vizinhança parcial de x deve ser avaliada.

2.2.4.2. Vizinhança parcial

Na vizinhança parcial (V_P), apenas uma porção da vizinhança total (V_T) de x é avaliada em cada iteração. O desafio agora passa a ser determinar quais soluções vizinhas serão avaliadas. O mecanismo empregado neste trabalho consiste em escolher esta vizinhança aleatoriamente. Neste caso, a escolha da nova variável básica ($X_{ij}=1$) em cada unidade de manejo é feita de forma aleatória. A variável candidata a deixar a base ($X_{ij}=0$) em cada unidade de manejo é aquela cujo valor é igual a um na solução x . No Quadro 2 é apresentado uma vizinhança parcial de x para o problema um, com quatro unidades de manejo e 12 variáveis de decisão geradas através do mecanismo de movimento descrito acima. Considerou-se uma vizinhança parcial (N_V) igual a cinco.

Quadro 2 - Vizinhança parcial de x de tamanho igual a cinco, gerada através do mecanismo de adicionar e remover aleatoriamente pares de variáveis X_{ij} em cada unidade de manejo (UM).

		UM (i)			1		2		3		4			Movimento				
		Variáveis			X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	Adc	Rem
		x			0	1	0	1	0	0	0	0	1	1	0	0	-	-
Vizinhança – $N_i(x, k)$	$N_1(x, k)$	0	1	0	0	1	0	0	0	0	1	1	1	0	0	X_{22}	X_{21}	
	$N_2(x, k)$	0	1	0	1	0	0	0	0	0	1	0	0	1	0	X_{43}	X_{41}	
	$N_3(x, k)$	1	0	0	1	0	0	0	0	0	1	1	1	0	0	X_{11}	X_{12}	
	$N_4(x, k)$	0	1	0	0	0	1	0	0	0	1	1	1	0	0	X_{23}	X_{21}	
	$N_5(x, k)$	0	1	0	1	0	0	0	0	1	0	1	0	1	0	X_{32}	X_{33}	

2.2.5. Função de avaliação

O mecanismo de movimento utilizado neste trabalho gera soluções viáveis segundo as restrições de singularidade. Em relação às demais restrições (produção

mínima e máxima), não há nenhuma garantia de viabilidade das soluções candidatas geradas. Ao invés de testar a viabilidade dessas soluções candidatas em relação às demais restrições do problema e aceitar apenas soluções viáveis segundo essas restrições, o mecanismo de avaliação utilizado pela metaheurística BT utiliza uma função de penalidade para guiar a escolha de soluções candidatas. A função de penalidade, $f_p(x)$, é obtida através de alterações na função objetivo, $f(x)$, do problema, da seguinte forma:

$$f_p(x) = f(x) - vp \times VT \quad (2)$$

$$f(x) = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} \quad (3)$$

em que:

$f_p(x)$ = valor da função objetivo penalizada para a solução x .

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

vp = penalização ($\$/m^3$) para cada unidade de produção violada.

VT^1 = violação total (m^3) das restrições de produção (mínima e máxima);

X_{ij} = variável de decisão que representa a j -ésima alternativa de manejo adotada na i -ésima unidade de manejo.

C_{ij} = é o valor presente líquido total (\$) de cada unidade de manejo i , colhida segundo a alternativa j ;

M = número total de unidades de manejo;

N = número total de alternativas de manejo na i -ésima unidade de manejo;

De acordo com a expressão (2), quanto maior o valor da violação total (VT) da solução candidata, maior será a penalidade atribuída a esta solução e menor será o seu desempenho em relação à função objetivo penalizada ($f_p(x)$), o que contribui para desencorajar a escolha dessa solução. Quando a violação total é nula ($VT=0$) os valores de $f_p(x)$ e $f(x)$ se equivalem.

2.2.6. Memória de curto prazo e restrições tabu

¹ Obtido conforme demonstrado no item 2.3.4.1. do capítulo 4.

O objetivo da restrição tabu é proibir movimentos que levem à soluções recentemente visitadas, utilizando-se algum mecanismo de controle. A BT lida com este problema incorporando uma estrutura de memória que proíbe ou penaliza certos movimentos de retornar a uma solução recentemente visitada. Os movimentos proibidos são chamados de *movimento tabu*. Ao intervalo definido de k iterações no qual um movimento permanece tabu denomina-se de *permanência* ($|T|$). Esta técnica de memória, conhecida como memória de curto prazo, uma vez que contém as soluções mais recentemente visitadas, prevenindo ciclos menores ou iguais a $|T|$.

Há varias formas de incorporar uma restrição tabu. Para o problema em foco, devido a natureza das restrições de singularidade (assinalamento de um único regime de manejo, X_{ij} , em cada talhão), a imposição de restrição tabu torna-se relativamente simples.

Seja X_{ij}^{Adc} a variável adicionada na melhor solução, $N_i^*(x,k)$. A restrição tabu imposta proíbe X_{ij}^{Adc} de ser removida (*Rem*) durante um período $|T|$ de iterações. Portanto, os movimentos tabu são todos aqueles que resultarem na remoção de X_{ij}^{Adc} durante $|T|$ iterações. Na prática, o controle da restrição tabu pode ser feito para cada unidade de manejo, uma vez que, proibir uma variável de decisão X_{ij}^{Adc} de ser removida nas próximas k iterações ($|T|$) significa proibir qualquer movimento que resulte da troca de variáveis na respectiva unidade de manejo, já que a restrição de singularidade atribui a cada unidade de manejo uma única variável de decisão com valor igual a um. Portanto, o número de soluções candidatas em um dado instante depende do número de unidades de manejo não tabu. Seja M_{NT} o número de unidades de manejo classificadas como não tabu em uma dada iteração. O tamanho da vizinhança em uma dada iteração é função dos movimentos derivados das unidades de manejo classificadas como não tabu, assim:

$$Nv = \sum_{i=1}^{M_{NT}} (N_{iNT} - 1) \quad (4)$$

em que:

Nv = tamanho da vizinhança em um dado instante;

M_{NT} = número de unidades de manejo não tabu;

N_{iNT} = número de alternativas de manejo na i -ésima unidade de manejo não tabu.

O tamanho da vizinhança (N_v) é máximo ($N_v = V_t$) somente quando todos os movimentos são classificados como não tabu, o que em geral ocorre na primeira iteração. Se todas as unidades de manejo forem classificadas como tabu (neste caso, $M_{NT} = 0$), não haverá nenhum movimento disponível e, portanto, o tamanho da vizinhança será igual a zero.

2.2.7. Lista tabu e atributos do movimento

O tipo da estrutura das listas tabu depende do problema em questão e não existe nenhuma estrutura que seja melhor para todas as aplicações. Entretanto, algumas regras gerais devem ser levadas em consideração no momento em que se planeja a estrutura da lista. A grande preocupação em encontrar a melhor estrutura de lista deve-se a fatores como a eficiência computacional, o que reforça a tese de que cada problema deve ter sua estrutura de lista definida com base na experimentação, quando na ausência de indicadores de uma estrutura mais adequada. A lista tabu utilizada pelo algoritmo BT implementado neste trabalho armazena o intervalo de iterações ou as próximas iterações em que o movimento permanecerá como movimento tabu (Quadro 3).

Apesar de não serem armazenados na lista tabu propriamente dita, alguns atributos do movimento como: a variável adicionada (Adc), a variável removida (Rem), a configuração da solução (x), as soluções vizinhanças ($Ni(x,k)$), o valor da função objetivo penalizada ($fp_i(x)$), a função objetivo não penalizada ($f_i(x)$), a violação total da i -ésima solução (VT_i), dentre outros, são aspectos que conferem informações à BT (Quadro 3). A forma de cálculos, armazenamento e recuperação desses atributos podem ainda afetar a performance da BT.

A permanência tabu de um movimento define o tamanho da lista tabu ($|T|$). O tamanho da lista pode ser estático, sem alteração durante toda a busca, ou

dinâmico, variando de tamanho ao longo do processo de busca. Apesar de haver algumas regras práticas para determinação de $|T|$, existe um consenso na literatura, reportando sobre BT, que a definição do tamanho de $|T|$ e sua forma de operação (estática ou dinâmica) é ainda dependente de experimentação. Neste trabalho foram empregadas duas formas de permanência tabu, ou seja, a estática ($|Te|$) e a dinâmica ($|TD|$).

No tamanho estático o valor da permanência ($|Te|$) escolhido permanece inalterado durante toda a busca.

Quadro 3 - Lista tabu representando as próximas k iterações com o melhor movimento escolhido considerado tabu e alguns atributos do movimento da metaheurística BT

Iteração o k	Soluções	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	Adc	Rem	$f_i(x)$	$f_{pi}(x)$	$VT_i(x)$	$ T $
	x	0	0	1	0	0	1	0	1	0	0	0	1	-	-	-	-	-	-
Vizinhança - $N_i(x, k)$	$N_1(x, k)$	0	1	0	0	0	1	0	1	0	0	0	1	X_{12}	X_{13}	$f_1(x)$	$fp_1(x)$	$VT_1(x)$	3
	$N_2(x, k)$	1	0	0	0	0	1	0	1	0	0	0	1	X_{11}	X_{13}	$f_2(x)$	$fp_2(x)$	$VT_2(x)$	3
	$N_3(x, k)$	0	0	1	0	1	0	0	1	0	0	0	1	X_{22}	X_{23}	0
	$N_4(x, k)$	0	0	1	1	0	0	0	1	0	0	0	1	X_{21}	X_{23}	0
	$N_5(x, k)$	0	0	1	0	0	1	1	0	0	0	0	1	X_{31}	X_{32}				0
	$N_6(x, k)$	0	0	1	0	0	1	0	0	1	0	0	1	X_{33}	X_{32}				0
	$N_7(x, k)$	0	0	1	0	0	1	0	1	0	1	0	0	X_{42}	X_{43}	2
	$N_8(x, k)$	0	0	1	0	0	1	0	1	0	0	1	0	X_{41}	X_{43}	$f_8(x)$	$fp_8(x)$	$VT_8(x)$	2

No tamanho dinâmico ($|TD|$), o valor da permanência tabu varia ao longo da busca de acordo com uma função de distribuição uniforme, dentro de um intervalo de variação mínimo (t_{min}) e máximo (t_{max}), ou seja, $|TD| \in \text{uniforme}\{t_{min}, t_{max}\}$. O valor de $|TD|$ escolhido em uma dada iteração pode permanecer constante durante α iterações, quando um novo valor de permanência é selecionado pelo mesmo processo. A determinação dos extremos t_{min} e t_{max} da distribuição uniforme são parâmetros importantes para propiciar uma duração tabu balanceada, evitando ciclagem e degeneração da busca.

2.2.8. Critérios de aspiração

Pode ocorrer no processo de busca que a vizinhança, em um dado instante, seja vazia, ou seja, todos os movimentos candidatos sejam considerados tabu, onde a busca seria interrompida. Para contornar esta situação pode-se dominar o estado tabu dos movimentos para cancelá-los quando necessário e permitir que a busca prossiga. A BT implementada neste trabalho permite o cancelamento do estado tabu de um movimento, e desta forma libera a sua execução permitindo o prosseguimento da busca, utilizando-se os critérios de aspiração *default* e por objetivo. Antes, porém, de aplicar os critérios de aspiração, o mecanismo de busca necessita gerar uma nova vizinhança. Para tal, escolheu-se uma nova vizinhança tabu parcial (VP) ou total (Vt) para eleger, dentre os movimentos desta vizinhança, aquele que terá o seu estado tabu cancelado. Se nenhum desses movimentos melhorar o valor da melhor solução até então encontrada, o critério de aspiração *default* é usado para cancelar o estado tabu de um movimento com a condição tabu mais antiga. Havendo empate, o movimento escolhido será aquele que possuir um melhor valor da função objetivo. No critério de aspiração por objetivo, o estado tabu do movimento cancelado é aquele que corresponde ao movimento da vizinhança tabu com um valor da função objetivo melhor do que o valor da função objetivo da melhor solução até então encontrada.

2.2.9. Critério de parada

O algoritmo BT proposto utiliza o mesmo critério de parada definido para o algoritmo genético desenvolvido no capítulo 4, ou seja: após encontrar a primeira solução com violação total nula ($VT=0$), a BT executa um número de iterações adicionais (Δk) para tentar melhorar a solução encontrada. Assim, a BT tenta encontrar várias soluções viáveis, escolhendo aquela de melhor valor para a função objetivo ($f^*(x)$). Para interromper a corrida da BT após um número de iterações

sem encontrar uma solução viável para o problema, foi definido um número máximo de iterações (k_{max}).

2.3. Escolha dos parâmetros da BT

Os parâmetros da metaheurística Busca Tabu avaliados neste trabalho foram os tipos de permanência (estática e dinâmica), o tamanho da permanência estática e o tamanho da vizinhança. Conforme já demonstrado, a permanência dinâmica é função dos parâmetros t_{min} , t_{max} e α . Além desses parâmetros básicos, a metaheurística BT utiliza outros parâmetros acessórios como o valor da penalização (vp), para cálculos das penalidades; o número de interações adicionais (Δk), para melhorar a qualidade das soluções viáveis encontradas e o número máximo de iterações permitidas (k_{max}). A Figura 4 mostra uma janela da rotina computacional, com opções para escolha dos parâmetros utilizados em uma dada corrida da BT.

Um resumo dos parâmetros e seus respectivos valores utilizados no teste da BT é apresentado no Quadro 4. As combinações entre todos os níveis dos diferentes parâmetros gerou um ensaio com 32 diferentes combinações, representando um cenário ou tratamento diferente. Cada tratamento foi repetido 4 vezes, resultando em total 128 observações para cada problema ou um total de 512 observações.

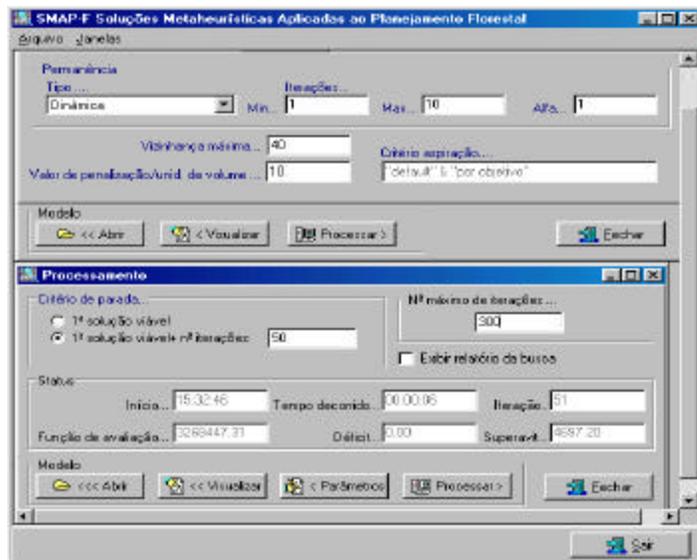


Figura 4– SMAP-F: interface computacional para definição dos parâmetros da BT.

Os valores de permanência estática e o limite superior da permanência dinâmica foram fixados como uma função do número de unidades de manejo (M), variável de acordo com o problema avaliado. Da mesma forma, o tamanho da vizinhança parcial é função da vizinhança total (Vt) que também varia de acordo com o problema.

Quadro 4 - Resumo dos parâmetros utilizados pela metaheurística Busca Tabu

Parâmetros	Níveis	Valores ou categorias dos parâmetros			
		Estática	Dinâmica		
Tipo de permanência	2				
Permanência estática ($ Te $)	4	0,25M	0,50M	0,75M	M
Tamanho máximo da vizinhança parcial (VP)	4	0,10Vt	0,20Vt	0,30Vt	0,40Vt
Limite mínimo para permanência dinâmica (t_{min})	1	1			
Limite máximo para permanência dinâmica (t_{max})	1	M			
Iterações para troca da permanência dinâmica (α)	1	1			
Número máximo de interação (k_{max})	1	300			
Interações adicionais (Δk)	1	50			

em que: M = número de unidades de manejo do problema e Vt é a vizinhança total do problema.

2.4. Avaliação da performance da BT

A performance do BT foi avaliada através da sua eficácia, calculada pela razão expressa em porcentagem entre o melhor valor da função objetivo obtida em uma corrida da BT e o ótimo matemático obtido pelo algoritmo exato *branch and bound*, assim:

$$Ef = \frac{f_{BT}}{f_o} \times 100 \quad (5)$$

em que:

Ef = eficácia (%);

f_{BT} = valor da melhor solução obtida em uma corrida da BT (\$);

f_o = valor da solução ótima obtida pelo algoritmo *branch and bound* (\$);

As medidas adequadas de eficiência dependem de estudos mais sofisticados, como a análise de complexidade de algoritmos, tais como complexidade de tempo e

de espaço. Neste trabalho, eficiência do BT foi avaliada utilizando-se apenas o tempo de processamento do algoritmo, o que permite apenas comparações relativas entre os métodos de solução, processadas utilizando-se os mesmos recursos computacionais.

2.5. Análise estatística dos resultados

As comparações entre os valores dos diferentes parâmetros foram efetuadas utilizando-se o procedimento estatístico proposto por LEITE e OLIVEIRA (2000) para comparar métodos analíticos. O procedimento proposto por estes autores, denominado teste L&O, utiliza uma metodologia que combina o teste F de GRAYBILL ($F(H_0)$), o teste t para o erro médio (t_e) e o coeficiente de correlação linear (r_{yij}) entre as variáveis. Além do teste L&O utilizou-se a estatística descritiva para auxiliar a análise dos resultados.

3. RESULTADOS E DISCUSSÃO

3.1. Ilustração de uma corrida da BT

No Quadro 5 é apresentado uma ilustração de uma corrida da heurística BT. Para tal utilizou-se o problema um, descrito no item 2.1 do capítulo quatro, com 12 variáveis de decisão, 4 restrições de singularidade e 4 restrições de produção mínima. O objetivo desse problema consistiu na maximização do valor presente líquido. A escolha do melhor movimento, em cada iteração foi feita utilizando-se a função objetivo penalizada ($f_p(x)$). Considerou-se uma permanência tabu estática ($|Te|$) igual a dois ($|Te| = 2$) e a exploração da vizinhança total. No método das penalidades, cada unidade de violação total (VT) é penalizada com um valor de penalidade (vp). Neste exemplo, assumiu-se este valor como sendo igual a dez ($vp=10$).

O procedimento inicia-se com uma solução inicial (x_0) gerada aleatoriamente. A partir de solução inicial ($x:=x_0$) gera-se a vizinhança total de x ,

ou seja, $N_i(x, k)$ através do mecanismo de movimento *Adc/Rem* determinístico (Quadro 5). Uma vez que o mecanismo de busca deve selecionar o movimento com maior $fp(x)$, tem-se então que a melhor solução vizinha na iteração um ($k=1$) é $N_5^*(x,1)$ com um valor de $fp(x)$ igual a \$ 333.000,00 e uma violação total (VT) de 10.000,00 m³. A solução $N_5^*(x, 1)$ é obtida pela adição de X_{32} e remoção de X_{33} . Inicialmente, a permanência estática ($|Te|$) é nula para todos os movimentos, o que significa que a lista tabu encontra-se vazia e, portanto, nenhum movimento é classificado como tabu.

Como o procedimento de busca consiste em escolher a melhor solução vizinha e, a partir dela, obter uma nova vizinhança, na iteração dois tem-se $x:=N_5^*(x, 1)$, e uma nova vizinhança para x é gerada pelo mecanismo de movimento *Adc/Rem*. Na iteração dois, a melhor solução vizinha x é $N_3^*(x, 2)$, com $fp(x)$ igual a \$ 321.000, 00 e VT igual a 10.000 m³, que consiste em adicionar a variável X_{22} e remover X_{23} . Note que, a exemplo do raciocínio anterior, na iteração em que a troca dos pares de variáveis é efetuada, $|Te|$ é nulo para as respectivas variáveis X_{22} adicionada (*Add*) e X_{23} removida (*Rem*), uma vez que $|Te|$ está definido para as próximas k iterações. Na iteração dois, existem dois movimentos classificados como tabu, que correspondem aos movimentos para geração das soluções $N_5(x,2)$ e $N_6(x,2)$. A primeira solução, $N_5(x,2)$, é obtida pela adição de X_{31} e remoção de X_{32} , enquanto a segunda solução, $N_6(x,2)$, é obtida pela adição de X_{33} e remoção de X_{32} . Como X_{32} é classificado como tabu ($|Te| = 2$), estes dois movimentos estão proibidos (tabu) de serem executados neste intervalo, uma vez que resultaria na remoção de X_{32} .

A cada iteração, as variáveis básicas ($X_{ij} = 1$), candidatas a deixarem a base, são mostradas na coluna *Rem*, enquanto as variáveis não básicas ($X_{ij} = 0$), candidatas a entrarem na base, estão na coluna *Adc*.

Um aspecto interessante da BT é o efeito da restrição tabu no tamanho da vizinhança. Considerando T como a lista tabu, contendo as $|Te|$ soluções mais recentemente visitadas, há uma redução no tamanho da vizinhança à medida em que a lista tabu é incrementada. Inicialmente, na iteração $k = 1$, tem-se o tamanho máximo da vizinhança, no exemplo, igual a oito ($Vt = 8$) (Quadro 5). O tamanho da

vizinhança depende do número de unidades manejo não tabu, uma vez que, para o problema com variáveis de decisão na forma binária, proibir a remoção de uma variável recentemente adicionada, resulta na proibição de qualquer movimento na respectiva unidade de manejo. Para o exemplo em foco, na iteração dois, tem-se três unidades de manejo não tabu, o que pelo procedimento de geração de movimento adotado, resulta em um total de seis movimentos ou soluções não tabu. Nas demais iterações o tamanho da vizinhança (para $|Te| = 2$) é sempre igual a quatro, já que existe em todas elas, quatro movimentos classificados como tabu. Os movimentos tabu foram gerados, propositalmente, com o objetivo de ilustração. Na prática, os movimentos tabu são gerados somente nas situações onde a vizinhança é vazia, ou seja, todos os movimentos são classificados como tabu. Neste caso, para aplicar os critérios de aspiração por *default* ou por objetivo, cancela-se temporariamente os estados tabu dos movimentos e gera-se uma nova vizinhança de x . O fato de não calcular os movimentos tabu em todas as iterações, resulta em economia de memória e tempo de processamento do algoritmo.

Na Figura 5 são apresentados os valores para a função objetivo $f(x)$ e seus respectivos valores penalizados ($fp(x)$), para as dez iterações apresentadas no Quadro 4. De acordo com esta figura, a BT permite piora na função objetivo, ou seja, aceita-se movimentos de pior qualidade. Esta característica da BT é uma das suas propriedades que permite a fuga de ótimos locais. Ao aceitar uma solução de pior qualidade, cria-se a oportunidade de direcionar a busca para regiões mais promissoras e, assim, encontrar uma solução de melhor qualidade. A aceitação de uma solução de pior qualidade ocorre devido ao efeito do parâmetro permanência que proíbe, temporariamente, uma solução tabu de melhor qualidade ser novamente revisitada, e desta forma evita a ciclagem ou degeneração da busca. Na iteração nove (Quadro 5, Figura 5) tem-se a igualdade dos valores de $f(x)$ e $fp(x)$, correspondente à um retorno líquido total de \$ 338.000,00. Neste ponto, tem-se a violação total nula e corresponde ao ponto onde a busca é interrompida pelo critério de parada ($VT=0$).

A solução encontrada pelo procedimento de BT corresponde à solução ótima para o problema, obtida pelo algoritmo de *branch and bound*. Entretanto,

para problemas maiores e, ou com espaço de soluções mais complexos, dificilmente será encontrada uma solução ótima para o problema. O que pode ocorrer, normalmente, é encontrar muitas soluções viáveis, no caso com VT nula. Nestes casos, o critério de parada pode ser alterado para permitir o prosseguimento de busca onde diversas soluções com défices nulos podem ser encontradas, sendo escolhida aquela solução com maior valor para a função objetivo. Com o objetivo de ilustrar a próxima solução gerada, foi efetuada mais uma iteração ($k = 10$) após o critério de parada ter sido atendido.

O método das penalidades utilizado neste trabalho representa, na prática, uma das táticas básicas de mecanismos de diversificação da BT. A idéia desses mecanismos é variar entre regiões viáveis e inviáveis, manipulando a função objetivo, através de penalidades ou incentivos, ou alterando movimentos construtivos (melhora da função objetivo) e destrutivo (piora da função objetivo) (GLAZAR, 2000). Uma das vantagens desses mecanismos é que o movimento direcionado para fora da fronteira de viabilidade e seu retorno em diferentes direções abre oportunidades para melhorar as soluções que não seriam obtidas se a busca ficasse confinada. De acordo com o exemplo ilustrativo, o método das penalidades manipulou as oito primeiras soluções inviáveis até que uma solução viável pudesse ser encontrada na iteração nove (Quadro 5), funcionando, portanto, como uma estratégia de diversificação bastante eficiente.

Quadro 5 – Ilustração de uma corrida da metaheurística Busca Tabu para um valor de permanência estática igual a dois ($|Te|=2$)

$k=1$	Soluções	X_1	X_1	X_1	X_2	X_2	X_2	X_3	X_3	X_3	X_4	X_4	X_4	Adc	Rem	Te	$f(x)$	VT	$fp(x)$	
		1	2	3	1	2	3	1	2	3	1	2	3							
	$x := x_0$	0	0	1	0	0	1	0	0	1	0	0	1	-	-	-	448000	20000	24800 0	
Vizinhança – $N_i(x, k)$	$N_1(x, 1)$	0	1	0	0	0	1	0	0	1	0	0	1	X_{12}	X_{13}	0	428000	10400	32400 0	
	$N_2(x, 1)$	1	0	0	0	0	1	0	0	1	0	0	1	X_{11}	X_{13}	0	408000	12000	28800 0	
	$N_3(x, 1)$	0	0	0	0	1	0	0	0	1	0	0	1	X_{22}	X_{23}	0	356000	20000	15600 0	
	$N_4(x, 1)$	0	0	1	1	0	0	0	0	1	0	0	1	X_{21}	X_{23}	0	418000	11600	30200 0	
	$N_5^*(x, 1)$	0	0	1	0	0	1	0	1	0	0	0	1	X_{32}	X_{33}	0	433000	10000	33300 0	
	$N_6(x, 1)$	0	0	1	0	0	1	1	0	0	0	0	1	X_{31}	X_{33}	0	418000	10000	31800 0	
	$N_7(x, 1)$	0	0	1	0	0	1	0	0	1	0	1	0	X_{42}	X_{43}	0	408000	20000	20800 0	
	$N_8(x, 1)$	0	0	1	0	0	1	0	0	1	1	0	0	X_{41}	X_{43}	0	368000	10000	26800 0	
$k=2$	$x := N_5^*(x, 1)$	0	0	1	0	0	1	0	1	0	0	0	1	-	-	-	433000	10000	33300 0	
Vizinhança – $N_i(x, k)$	$N_1(x, 2)$	0	1	0	0	0	1	0	1	0	0	0	1	X_{12}	X_{13}	0	413000	20000	21300 0	
	$N_2(x, 2)$	1	0	0	0	0	1	0	1	0	0	0	1	X_{11}	X_{13}	0	393000	12000	27300 0	
	$N_3^*(x, 2)$	0	0	1	0	1	0	0	1	0	0	0	1	X_{22}	X_{23}	0	421000	10000	32100 0	
	$N_4(x, 2)$	0	0	1	1	0	0	0	1	0	0	0	1	X_{21}	X_{23}	0	403000	10000	30300 0	
	$N_5(x, 2)$	0	0	1	0	0	1	1	0	0	0	0	1	X_{31}	X_{32}	2	418000	10000	31800 0	
	$N_6(x, 2)$	0	0	1	0	0	1	0	0	1	0	0	1	X_{33}	X_{32}	2	448000	20000	24800 0	
	$N_7(x, 2)$	0	0	1	0	0	1	0	1	0	1	0	0	X_{41}	X_{43}	0	353000	10000	25300 0	
	$N_8(x, 2)$	0	0	1	0	0	1	0	1	0	0	1	0	X_{42}	X_{43}	0	393000	10000	29300 0	
$k=3$	$x := N_3^*(x, 1)$	0	0	1	0	1	0	0	1	0	0	0	1	-	-	-	421000	10000	32100 0	
Vizinha	$N_1(x, 3)$	0	1	0	0	1	0	0	1	0	0	0	1	X_{12}	X_{13}	0	401000	10000	30100 0	
	$N_2^*(x, 3)$	1	0	0	0	1	0	0	1	0	0	0	1	X_{11}	X_{13}	0	381000	2000	36100 0	

	$N_3(x, 3)$	0	0	1	1	0	0	0	1	0	0	0	1	X_{21}	X_{22}	2	403000	10000	30300 0
	$N_4(x, 3)$	0	0	1	0	0	1	0	1	0	0	0	1	X_{23}	X_{22}	2	433000	10000	33300 0
	$N_5(x, 3)$	0	0	1	0	1	0	1	0	0	0	0	1	X_{31}	X_{32}	1	406000	10000	30600 0
	$N_6(x, 3)$	0	0	1	0	1	0	0	0	1	0	0	1	X_{33}	X_{32}	1	436000	20000	23600 0
	$N_7(x, 3)$	0	0	1	0	1	0	0	1	0	1	0	0	X_{41}	X_{43}	0	341000	20000	14100 0
	$N_8(x, 3)$	0	0	1	0	1	0	0	1	0	0	1	0	X_{42}	X_{43}	0	381000	20000	18100 0
$k=4$	$x:=N_2^*(x, 3)$	1	0	0	0	1	0	0	1	0	0	0	1	-	-	-	381000	2000	36100 0
<i>Vizinhança – $N_i(x, k)$</i>	$N_1(x, 4)$	0	1	0	0	1	0	0	1	0	0	0	1	X_{12}	X_{11}	2	401000	10000	30100 0
	$N_2(x, 4)$	0	0	1	0	1	0	0	1	0	0	0	1	X_{13}	X_{11}	2	421000	10000	32100 0
	$N_3(x, 4)$	1	0	0	1	0	0	0	1	0	0	0	1	X_{21}	X_{22}	1	363000	12000	24300 0
	$N_4(x, 4)$	1	0	0	0	0	1	0	1	0	0	0	1	X_{23}	X_{22}	1	393000	12000	27300 0
	$N_5(x, 4)$	1	0	0	0	1	0	1	0	0	0	0	1	X_{31}	X_{32}	0	366000	10000	26600 0
	$N_6^*(x, 4)$	1	0	0	0	1	0	0	0	1	0	0	1	X_{33}	X_{32}	0	396000	12000	27600 0
	$N_7(x, 4)$	1	0	0	0	1	0	0	1	0	1	0	0	X_{41}	X_{43}	0	301000	12000	18100 0
	$N_8(x, 4)$	1	0	0	0	1	0	0	1	0	0	1	0	X_{43}	X_{43}	0	341000	12000	22100 0

Continua

Quadro 5 – Continuação...

$k=5$	Soluções	X_1	X_1	X_1	X_2	X_2	X_2	X_3	X_3	X_3	X_4	X_4	X_4	Adc	Rem	$ Te $	$f(x)$	VT	$fp(x)$
	$x:=N_6^*(x, 4)$	1	0	0	0	1	0	0	0	1	0	0	1	-	-	-	396000	12000	27600 0
<i>Vizinha</i>	$N_1(x, 5)$	0	1	0	0	1	0	0	0	1	0	0	1	X_{12}	X_{11}	1	416000	10400	31200 0
	$N_2(x, 5)$	0	0	1	0	1	0	0	0	1	0	0	1	X_{13}	X_{11}	1	436000	20000	23600 0

	$N_3^*(x, 5)$	1	0	0	1	0	0	0	0	1	0	0	1	X_{21}	X_{22}	0	378000	3600	34200 0
	$N_4(x, 5)$	1	0	0	0	0	1	0	0	1	0	0	1	X_{23}	X_{22}	0	408000	12000	28800 0
	$N_5(x, 5)$	1	0	0	0	1	0	1	0	0	0	0	1	X_{31}	X_{33}	2	366000	10000	26600 0
	$N_6(x, 5)$	1	0	0	0	1	0	0	1	0	0	0	1	X_{32}	X_{33}	2	381000	2000	36100 0
	$N_7(x, 5)$	1	0	0	0	1	0	0	0	1	1	0	0	X_{42}	X_{43}	0	316000	12000	19600 0
	$N_8(x, 5)$	1	0	0	0	1	0	0	0	1	0	1	0	X_{41}	X_{43}	0	356000	22000	13600 0
$k=6$	$x:=N_3^*(x, 5)$	1	0	0	1	0	0	0	0	1	0	0	1	-	-	-	378000	3600	34200 0
Vizinhança – $N_i(x, k)$	$N_1(x, 6)$	0	1	0	1	0	0	0	0	1	0	0	1	X_{12}	X_{11}	0	398000	10000	29800 0
	$N_2^*(x, 6)$	0	0	1	1	0	0	0	0	1	0	0	1	X_{13}	X_{11}	0	418000	11600	30200 0
	$N_3(x, 6)$	1	0	0	0	1	0	0	0	1	0	0	1	X_{22}	X_{21}	2	396000	12000	27600 0
	$N_4(x, 6)$	1	0	0	0	0	1	0	0	1	0	0	1	X_{23}	X_{21}	2	408000	12000	28800 0
	$N_5(x, 6)$	1	0	0	1	0	0	0	1	0	0	0	1	X_{32}	X_{33}	1	363000	12000	24300 0
	$N_6(x, 6)$	1	0	0	1	0	0	1	0	0	0	0	1	X_{31}	X_{33}	1	348000	11600	23200 0
	$N_7(x, 6)$	1	0	0	1	0	0	0	0	1	1	0	0	X_{41}	X_{43}	0	298000	12000	17800 0
	$N_8(x, 6)$	1	0	0	1	0	0	0	0	1	0	1	0	X_{42}	X_{43}	0	338000	13600	20200 0
$k=7$	$x:=N_2^*(x, k)$	0	0	1	1	0	0	0	0	1	0	0	1	-	-	-	418000	11600	30200 0
Vizinhança – $N_i(x, k)$	$N_1(x, 7)$	1	0	0	1	0	0	0	0	1	0	0	1	X_{11}	X_{13}	2	378000	3600	34200 0
	$N_2(x, 7)$	0	1	0	1	0	0	0	0	1	0	0	1	X_{12}	X_{13}	2	398000	10000	29800 0
	$N_3(x, 7)$	0	0	1	0	1	0	0	0	1	0	0	1	X_{22}	X_{21}	1	436000	20000	23600 0
	$N_4(x, 7)$	0	0	1	0	0	1	0	0	1	0	0	1	X_{23}	X_{21}	1	448000	20000	24800 0
	$N_5^*(x, 7)$	0	0	1	1	0	0	1	0	0	0	0	1	X_{31}	X_{33}	0	388000	1600	37200 0
	$N_6(x, 7)$	0	0	1	1	0	0	0	1	0	0	0	1	X_{32}	X_{33}	0	403000	10000	30300 0
	$N_7(x, 7)$	0	0	1	1	0	0	0	0	1	1	0	0	X_{41}	X_{43}	0	338000	20000	13800 0
	$N_8(x, 7)$	0	0	1	1	0	0	0	0	1	0	1	0	X_{42}	X_{43}	0	378000	21600	16200 0

$k=8$	$x:=N_5^*(x, 7)$	0	0	1	1	0	0	1	0	0	0	0	1	-	-	-	388000	1600	37200 0
<i>Vizinhança – $N_i(x, k)$</i>	$N_1(x, 8)$	0	1	0	1	0	0	1	0	0	0	0	1	X_{12}	X_{13}	1	368000	10000	26800 0
	$N_2(x, 8)$	1	0	0	1	0	0	1	0	0	0	0	1	X_{11}	X_{13}	1	348000	11600	23200 0
	$N_3(x, 8)$	0	0	1	0	1	0	1	0	0	0	0	1	X_{22}	X_{21}	0	406000	10000	30600 0
	$N_4^*(x, 8)$	0	0	1	0	0	1	1	0	0	0	0	1	X_{23}	X_{21}	0	418000	10000	31800 0
	$N_5(x, 8)$	0	0	1	1	0	0	0	1	0	0	0	1	X_{32}	X_{31}	2	403000	10000	30300 0
	$N_6(x, 8)$	0	0	1	1	0	0	0	0	1	0	0	1	X_{33}	X_{31}	2	418000	11600	30200 0
	$N_7(x, 8)$	0	0	1	1	0	0	1	0	0	0	1	0	X_{42}	X_{43}	0	348000	11600	23200 0
	$N_8(x, 8)$	0	0	1	1	0	0	1	0	0	1	0	0	X_{41}	X_{43}	0	308000	10000	20800 0

Continua

Quadro 5 – Continuação...

$k=9$	Soluções	X_1	X_1	X_1	X_2	X_2	X_2	X_3	X_3	X_3	X_4	X_4	X_4	X_4	Adc	Rem	Te	$f(x)$	VT	$fp(x)$
		1	2	3	1	2	3	1	2	3	1	2	3							
	$x:=N_4^*(x, 8)$	0	0	1	0	0	1	1	0	0	0	0	1	-	-	-	41800 0	10000	31800 0	
<i>Vizinhança – $N_i(x, k)$</i>	$N_1(x, 9)$	0	1	0	0	0	1	1	0	0	0	0	1	X_{12}	X_{13}	0	39800 0	10400	29400 0	
	$N_2(x, 9)$	1	0	0	0	0	1	1	0	0	0	0	1	X_{11}	X_{13}	0	37800 0	20000	17800 0	
	$N_3(x, 9)$	0	0	1	1	0	0	1	0	0	0	0	1	X_{21}	X_{23}	2	38800 0	1600	37200 0	
	$N_4(x, 9)$	0	0	1	0	1	0	1	0	0	0	0	1	X_{22}	X_{23}	2	40600 0	10000	30600 0	
	$N_5(x, 9)$	0	0	1	0	0	1	0	1	0	0	0	1	X_{32}	X_{31}	1	43300 0	10000	33300 0	
	$N_6(x, 9)$	0	0	1	0	0	1	0	0	1	0	0	1	X_{33}	X_{31}	1	44800 0	20000	24800 0	
	$N_7^*(x, 9)$	0	0	1	0	0	1	1	0	0	1	0	0	X_{41}	X_{43}	0	33800 0	0	33800 0	
	$N_8(x, 9)$	0	0	1	0	0	1	1	0	0	0	1	0	X_{42}	X_{43}	0	37800 0	10000	27800 0	

$k=10$	$x := N_7^*(x, 9)$	0	0	1	0	0	1	1	0	0	1	0	0	-	-	-	33800	0	33800
Vizinhança – $N_i(x, k)$	$N_1(x, 10)$	0	1	0	0	0	1	1	0	0	1	0	0	X_{12}	X_{13}	0	31800	10000	21800
	$N_2(x, 10)$	1	0	0	0	0	1	1	0	0	1	0	0	X_{11}	X_{13}	0	29800	10000	19800
	$N_3(x, 10)$	0	0	1	1	0	0	1	0	0	1	0	0	X_{21}	X_{23}	1	30800	10000	20800
	$N_4(x, 10)$	0	0	1	0	1	0	1	0	0	1	0	0	X_{22}	X_{23}	1	32600	10000	22600
	$N_5(x, 10)$	0	0	1	0	0	1	0	1	0	1	0	0	X_{32}	X_{31}	0	35300	10000	25300
	$N_6^*(x, 10)$	0	0	1	0	0	1	0	0	1	1	0	0	X_{33}	X_{31}	0	36800	10000	26800
	$N_7(x, 10)$	0	0	1	0	0	1	1	0	0	0	1	0	X_{42}	X_{41}	2	37800	10000	27800
	$N_8(x, 10)$	0	0	1	0	0	1	1	0	0	0	0	1	X_{43}	X_{41}	2	41800	10000	31800

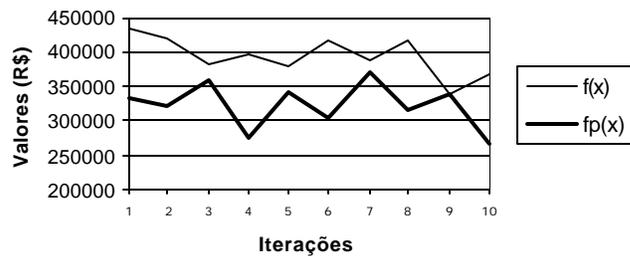


Figura 5 – Valores da função objetivo penalizada ($fp(x)$) e não penalizada ($f(x)$) para uma corrida da BT.

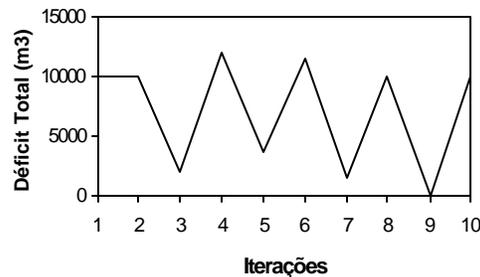


Figura 6 – Violação total nas restrições de produção mínima para uma corrida da BT.

3.2. Avaliação dos parâmetros e medidas de performance da BT

Para os problemas testados, houve diferenças estatísticas significativas, pelo teste de L&O, entre alguns níveis ou categorias dos parâmetros da BT, avaliados a um nível de significância de 1% (Quadro 6).

As medidas de eficácia da BT e respectivas frequências para os quatro diferentes valores de permanência estática estão apresentadas nas Figuras 7 (a, b). Os valores de permanência correspondentes a 0,25M e 0,50M das unidades de manejo do problema propiciaram uma maior eficácia na BT quando comparados aos valores correspondentes a 0,75M e M. Esta tendência é reforçada pelos resultados da maior frequência nas classes superiores de eficácia para os valores de 0,25M e 0,50M (Figura 7b). A permanência é um dos principais parâmetros da BT, podendo afetar a sua performance. Valores baixos podem diminuir a diversidade da busca devido a ciclagem e degeneração, podendo levar a estacionamento em ótimos locais pela dificuldade de exploração de novas soluções. Valores altos apesar de forçarem a pesquisa de novas soluções, funcionando como um artifício para fuga de ótimos locais, podem também diminuir a diversidade da busca devido à diminuição no tamanho da vizinhança, reduzindo as chances de encontrar uma solução de melhor qualidade. Este argumento parece ter sido o que mais afetou a performance da BT, ou seja, maiores valores de permanência resultaram em uma menor eficácia da BT devido à redução da vizinhança com conseqüente reflexo na performance da BT.

Não houve diferenças estatísticas significativas, pelo teste de L&O a um nível de 1% de probabilidade, entre a permanência estática e dinâmica avaliadas (Quadro 6). Cabe ressaltar que a permanência dinâmica foi comparada com os melhores valores de permanência estática (0,25M e 0,50M). Além disso, a permanência dinâmica adotada neste trabalho considerou um intervalo de variação com limite mínimo igual a um ($t_{min}=1$), limite máximo igual ao número de unidades de manejo do problema ($t_{max}=M$) e a escolha de um novo valor de permanência a cada nova iteração ($\alpha=1$). Na Figura 7c é apresentada a eficácia da BT para estes dois parâmetros. Como não houve diferenças estatísticas significativas entre a

permanência dinâmica e as melhores configurações de permanência estática, pode-se inferir que nas situações onde não haja nenhum indicativo do melhor valor de permanência estática, a permanência dinâmica possa ser adotada, principalmente devido à maior flexibilidade na definição deste parâmetro. É necessário investigar o efeito da permanência dinâmica para outros valores de t_{min} , t_{max} e α , principalmente devido à possibilidade melhorias na eficácia da BT para tais combinações.

Em relação ao tamanho da vizinhança parcial máxima, apenas o tamanho igual a $0,1Vt$ diferiu dos demais valores pelo teste de L&O a um nível de 1% de probabilidade (Quadro 6). Os efeitos do tamanho da vizinhança parcial máxima na eficácia da BT são apresentados nas Figuras 7(d, e). Maiores valores de vizinhança parcial máximas tenderam a apresentar maiores valores de eficácia (Figura 7d), resultado reforçado pela maior frequência nas maiores classes de eficácia (Figura 7e).

O tamanho da vizinhança parcial máxima varia com a vizinha total (Vt) de cada problema. No Quadro 7 é apresentado a vizinhança total dos quatro problemas avaliados. Os valores apresentados dependem do tipo de movimento empregado (neste trabalho derivados por mecanismo de adicionar e remover em cada unidade de manejo pares de variáveis de decisão, X_{ij}). Outra observação importante é que a vizinhança total em um dado instante depende do número de unidades de manejo classificadas como não tabu. Portanto, a vizinhança total é máxima quando todas as unidades de manejo forem não tabu, o que em geral, ocorre na primeira iteração. O tamanho da vizinhança é nulo quando todas as unidades de manejo são classificadas como tabu, quando a busca seria interrompida devido a indisponibilidade de novos movimentos.

Quadro 6 - Teste de L&O² para comparar os efeitos dos parâmetros da metaheurística Busca Tabu

Valores da permanência estática ³		F(H0)	t	r
0,25M	0,50M	n.s	n.s	sim

² Regra decisória descrita no item 2.3.12. do capítulo 4.

³ Valores fracionados foram arredondados para inteiros imediatamente superiores

0,25M	0,75M	n.s	*	sim
0,25M	M	n.s	*	sim
0,50M	0,75M	n.s	*	sim
0,50M	M	n.s	*	sim
0,75M	M	n.s	n.s	sim
Tipo de permanência		F(H0)	t	r
Estática	Dinâmica	n.s	n.s	sim
Tamanho da vizinhança ³		F(H0)	t	r
0,10Vt	0,20Vt	n.s	*	sim
0,10Vt	0,30Vt	n.s	*	sim
0,10Vt	0,40Vt	n.s	*	sim
0,20Vt	0,30Vt	n.s	n.s	sim
0,20Vt	0,40Vt	n.s	n.s	sim
0,30Vt	0,40Vt	n.s	n.s	sim

em que: M = número de unidades de manejo do problema e Vt é a vizinhança total não tabu do problema.

Para evitar que a busca seja interrompida pela falta de movimentos candidatos, a BT permite o cancelamento do estado tabu de alguns movimentos para permitir o prosseguimento da busca. O mecanismo que permite o cancelamento do estado tabu dos movimentos é o critério de aspiração. Neste trabalho foi utilizado dois critérios de aspiração: *default* e por objetivo. O critério de aspiração *default* permitiu o cancelamento do estado tabu do movimento com a condição tabu mais antiga, naquelas situações onde nenhum movimento candidato melhorava o valor da melhor solução até então encontrada. Por outro lado, o critério de aspiração por objetivo, permitiu a escolha de um novo movimento se este resultava em uma melhoria da função objetivo da melhor solução até então encontrada. Na estratégia de busca empregada, os dois critérios são mutuamente exclusivos, ou seja, a escolha do critério empregado depende da avaliação de uma nova vizinhança gerada pelo cancelamento temporário do estado tabu de todos os movimentos disponíveis. O tamanho da vizinhança neste instante é igual ao valor da vizinhança parcial máxima escolhido para uma dada corrida.

A eficácia da BT para os diferentes valores e categorias de parâmetros estudados oscilou entre os valores de 85 e 100% do ótimo matemático obtido pelo algoritmo *branch and bound* (Figuras 7a, 7c e 7d). Entretanto, ao se utilizar os melhores valores para os seus parâmetros, a BT apresentou uma melhor eficácia

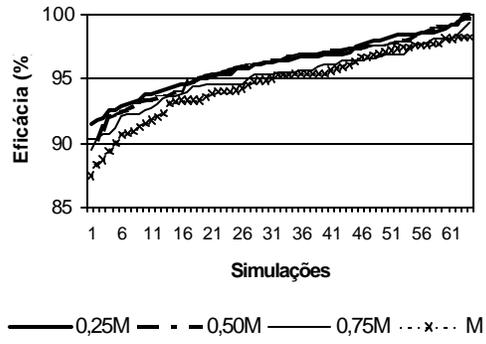
com média de 95,97%, valor mínimo igual a 90,39%, valor máximo igual a 98,84% do ótimo matemático, com um coeficiente de variação de 2,48%.

Quadro 7 – Vizinhança total de cada problema, considerando o mecanismo de movimento de adicionar e remover em cada unidade de manejo pares de variáveis de decisão

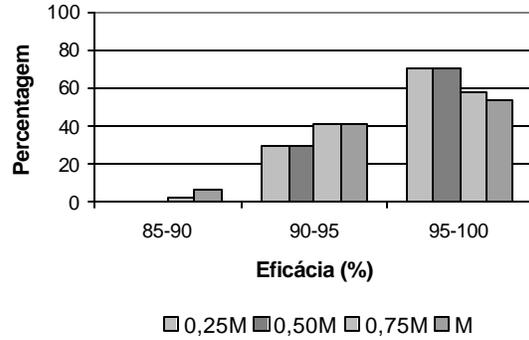
Problema	Vizinhança total (Vt)
2	83
3	196
4	196
5	384

A principal desvantagem de se ter resultados melhores de eficácia da BT com maiores valores de vizinhanças deve-se à depreciação no tempo computacional da busca com o aumento do tamanho da vizinhança (Quadro 8). Por exemplo, para o problema de maior porte (problema 5), ao se quadruplicar o tamanho da vizinhança, houve um aumento na mesma proporção no tempo computacional. Este aumento deve-se ao maior esforço exigido pela BT para gerar e avaliar um maior número de soluções candidatas com o aumento da vizinhança.

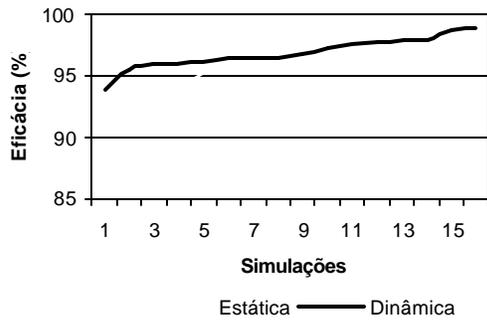
A eficiência da BT não foi avaliada rigorosamente nesse trabalho. Entretanto, os resultados computacionais preliminares da BT mostraram a superioridade desse método quando comparado ao algoritmo exato *branch and bound*. O tempo computacional gasto pelo algoritmo *branch and bound* cresce mais rapidamente com o tamanho do problema (número de variáveis), comparado à BT. No problema quatro com 423 variáveis o algoritmo *branch and bound* consumiu um tempo duas vezes superior à BT (Figura 8).



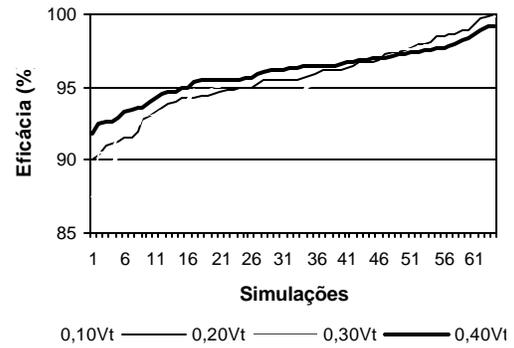
(a)



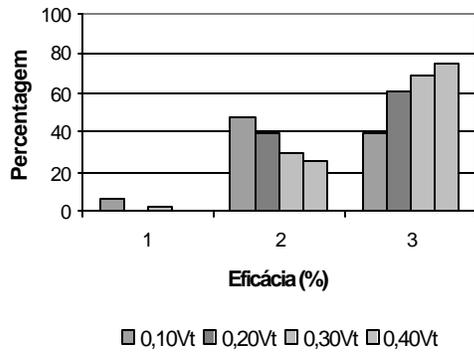
(b)



(c)



(d)



(e)

Figura 7 – Medidas de eficácia e respectivas freqüência para diferentes valores de permanência estática (a, b), tipos de permanência (c) e tamanhos de vizinhanças (d, e) para a metaheurística Busca Tabu.

Quadro 8 – Tempos médios (segundos) de processamento da BT para diferentes valores de percentagem da vizinhança parcial máxima para os quatro problemas avaliados

Problema	Tamanho da vizinhança parcial máxima			
	10% Vt	20% Vt	30% Vt	40% Vt
2	1	2	4	6
3	21	34	50	64
4	20	40	75	79
5	170	245	466	599

em que: Vt é a vizinhança total não tabu

Apesar das vantagens já obtidas, o tempo de processamento da BT deve ser melhorado. Melhorias na estrutura de armazenamento e recuperação de dados, mecanismos de cálculos mais eficientes para avaliações das restrições e função objetivo do problema, mecanismos para geração de movimentos e classificação tabu, são apenas alguns dos aspectos da BT que podem ser melhorados. Tais melhorias são vitais para aumentar a capacidade de processamento da BT em tempo computacionais viáveis, e assim viabilizar as suas aplicações em problemas de maior porte.

Outro aspecto que pode melhorar ainda mais a performance da BT, principalmente em relação a melhoria da sua eficácia, é a incorporação de outros artifícios da BT como a inclusão de estruturas de memória de longo prazo, aspecto ainda não explorado neste trabalho. A inclusão dessas estruturas de memória podem ser decisivas para o sucesso da BT em problemas maiores e, ou mais complexos. Além disso, a exemplo do algoritmo genético apresentado no capítulo anterior, outros fatores com a qualidade do critério para interrupção da busca podem afetar a performance da BT e merecem ser melhor investigados.

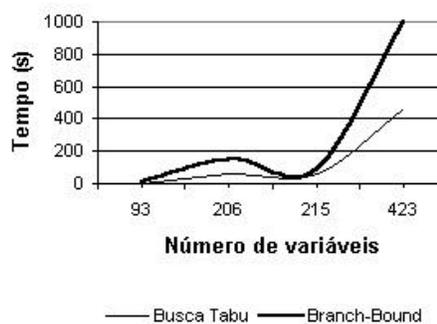


Figura 8– Tempo de solução obtido pela metaheurística Busca Tabu e o algoritmo *branch and bound*.

4. CONCLUSÕES

Este trabalho utilizou a metaheurística Busca Tabu (BT) para resolver problemas de planejamento florestal com restrições de inteireza. Foram testados cinco problemas contendo entre 12 e 423 variáveis de decisão, com o objetivo de maximizar a receita líquida, sujeitos às restrições de singularidade e produção mínima e máxima.

Um problema de pequeno porte (com 12 variáveis de decisão com restrições de singularidade e produção mínima) foi utilizado para ilustrar uma corrida da BT. Os demais problemas foram utilizados para avaliar a performance da BT para algumas variações em seus parâmetros.

A medida de eficácia, expressa pela relação entre o valor da função objetivo encontrada pela BT e o ótimo matemático obtido pelo algoritmo exata *branch and bound*, foi utilizada para avaliar a performance da BT.

Os diferentes parâmetros da Busca Tabu e seus respectivos valores avaliados foram estaticamente iguais pelo teste L&O, a um nível de 1% de probabilidade. Algumas configurações de parâmetros propiciaram uma melhor eficácia para a BT. Neste sentido, os valores de permanência estática fixada em 25 e 50% do número de unidade de manejo do problema, apresentara eficácias superiores aos demais valores (75 e 100%). A permanência dinâmica foi uma boa estratégia para melhorar a eficácia da BT já que apresentou resultados semelhantes aos melhores valores de permanência estática. Tamanhos maiores de vizinhança parcial (30 e 40% da vizinhança total) propiciaram maiores eficácia para a BT. Entretanto, a eficiência da BT, avaliada pelo consumo de tempo de processamento, sofreu uma depreciação para os maiores tamanhos de vizinhança.

Para os quatros problemas estudados, os melhores parâmetros da BT forneceram soluções com eficácia média de 95,97%, valor mínimo igual a 90,39%, valor máximo igual a 98,84% do ótimo matemático, com um desvio padrão de 2,48%. O tempo de processamento para encontrar uma boa solução para o problema

de maior porte, considerando uma vizinhança parcial de 30% da vizinhança total, foi duas vezes inferior ao principal algoritmo exato de solução (*branch and bound*).

A incorporação de outros artifícios da BT como a inclusão de estruturas de memória de longo prazo, aspecto ainda não explorados neste trabalho, pode contribuir ainda mais para o sucesso da BT em resolver problemas de planejamento florestal, principalmente os de grande porte e, ou de maior complexidade.

Aplicabilidade universal da BT, propiciada pela flexibilidade desses algoritmos em resolver problemas de diferentes domínios, cabendo apenas adaptá-los aos seus padrões, permite inferir que esta técnica pode ser utilizada para resolver uma variedade de outros importantes problemas de planejamento florestal, entre eles, aqueles que envolvem restrições de adjacência, roteamento de veículos para o transporte florestal, corte na indústria, seccionamento de toras etc.

Os objetivos desse trabalho foram alcançados. Entretanto, é necessário melhorar a capacidade da rotina computacional para viabilizar a aplicação da Busca Tabu em problemas maiores.

A melhor performance da Busca Tabu é dependente da escolha de uma boa configuração de seus parâmetros. A inexistência de conclusões seguras para a maioria dos problemas de planejamento florestal, em especial, os problemas maiores e, ou mais complexos, sugerem que essa escolha deve ser baseada na experimentação.

5. REFERÊNCIAS BIBLIOGRÁFICAS

BOSTON, K.; BETTINGER, P. An Analysis of Monte Carlo Integer Programming, Simulated Annealing, and Tabu Search Heuristics for Solving Spatial Harvest Scheduling Problems. Forest Science, Washington, 45(2):292–301, 1999.

GLAZAR, E. J. N. Análise de diferentes estratégias na aplicação da busca tabu ao problema de roteamento de veículos. Rio de Janeiro, RJ: UFRJ, 2000. 132p. Dissertação (Mestrado em Ciência em Engenharia de Produção) - Universidade Federal de do Rio de Janeiro, 2000.

GLOVER, F. Future paths for integer programming and links to artificial intelligence. Computers and Operational Research, 5, 533-549. 1986

- GLOVER, F. Heuristics for integer programming using surrogate constraints.** Decision Science, **8**, 1977. p.156-166.
- GLOVER, F.; LAGUNA, M.** Tabu search. **Kluwer academic publishers, Boston. 1997. 382p.**
- LAROZE, A.J.** A linear programming, tabu search method for solving forest-level bucking optimization problems. Forest Science, Washington, **45(1):108-116, 1999.**
- LAROZE, A.J.; GREBER, B.J. Using Tabu search to generate stand-level, rule-based bucking patterns. **Forest Science**, Washington, 43(2):157-169, 1997.
- LEITE, H.G. & OLIVEIRA, F.H.T. Statistical Procedure to Test the Identity of Analytical Methods. *Comm. Soil Sci. Plant Anal.*, 2001, 16p. (no prelo)
- RICHARDS, E.W.; GUNN, E.A.** A Model and Tabu Search Method to Optimize Stand Harvest and Road Construction Schedules. Forest Science, Washington, **46(2):188-203, 2000.**
- WERRA, D. HERTZ, A. Tabu search techniques: a tutorial and an application to neural networks. **OR Spektrum**, v.11, 131-141, 1989.

CAPÍTULO 6

METAHEURÍSTICA *SIMULATED ANNEALING* PARA SOLUÇÃO DE PROBLEMAS DE PLANEJAMENTO FLORESTAL COM RESTRIÇÕES DE INTEIREZA

RESUMO - Este capítulo teve como objetivo desenvolver e testar a metaheurística SA para a solução de problemas de gerenciamento florestal com restrições de inteireza. O algoritmo SA desenvolvido foi testado em cinco problemas, contendo entre 12 e 423 variáveis de decisão, sujeitos às restrições de singularidade, produção mínima e produção máxima periodicamente. Todos os problemas tiveram como objetivo a maximização do valor presente líquido. O algoritmo SA foi codificado em linguagem *delphi* 5.0 e os testes efetuados em um microcomputador AMD K6II 500 MHZ, com memória RAM de 64 MB e disco rígido de 15GB. A performance da SA foi avaliada de acordo com as medidas de eficácia e eficiência. Os diferentes valores ou categorias dos parâmetros da SA foram testados e comparados quanto aos seus efeitos na eficácia do algoritmo. A seleção da melhor configuração de parâmetros foi feita utilizando o teste L&O, a um nível de 1% de probabilidade e análises através de estatísticas descritivas. A melhor configuração de parâmetros propiciou à SA eficácia média igual a 95,36%, valor mínimo igual a 83,66%, valor máximo igual a 100% e coeficiente de variação igual a 3,18% do ótimo matemático obtido pelo algoritmo exato *branch and bound*. Para o problema de maior porte, a eficiência da SA foi dez vezes superior à eficiência do algoritmo exato *branch and bound*. A boa performance desta heurística reforçou as conclusões, tiradas em outros trabalhos, do seu enorme potencial para a solução de importantes problemas de

gerenciamento florestal de difícil solução pelos instrumentos computacionais da atualidade.

Palavras-chave: Gerenciamento florestal, heurísticas, *Simulated Annealing*.

METAHEURISTIC SIMULATED ANNEALING TO SOLVE FOREST PLANNING PROBLEM WITH INTEGER CONSTRAINTS

ABSTRACT – The objective of this chapter was to develop and test an algorithm based in Simulated Annealing (SA) metaheuristic to solve problems of forest management with integer constraints. The algorithm SA developed was tested in five problems containing between 12 and 423 decision variables, subjected to singularity constraints, minimum and maximum production periodically. All the problems had the objective of maximizing the net present value. SA was codified into *delphi* 5.0 language and the tests were performed in a microcomputer AMD K6II 500 MHZ, with RAM memory of 64 MB and hard disk of 15GB. The SA performance was evaluated according to the efficacy and efficiency measures. The different values or categories for the SA parameters were tested and compared in relation to their effects on the algorithm efficacy. The selection of the parameters' best configuration was performed by using the L&O test at 1% probability and analyses via descriptive statistics. The parameters' best configuration provided for SA average efficacy of 95,36%, minimum value equal to 83,66%, maximum value equal to 100%, with coefficient of variation being 3,18% of the mathematical optimum, obtained by the exact algorithm *branch and bound*. As for the larger problem, the efficiency of SA was ten times superior to the efficiency of the exact algorithm *branch and bound*. SA came out as a quite attractive new approach in forest management for solving important problems of difficult solution through the use of the current computational instruments.

Key-words: Forest Management, Heuristic and Simulated Annealing.

1. INTRODUÇÃO

A metaheurística *Simulated Annealing* (SA) foi derivada a partir de processos de recozimento sólidos desenvolvido por Metropolis *et al* (1953), citado por YOUSSEF (2001). A analogia com a otimização combinatória foi introduzida por KIRKPATRICK *et al.* (1983) e aperfeiçoada por CERNY (1985). O termo *annealing* refere-se a um processo resfriamento térmico que começa pela liquidação de um cristal, a uma alta temperatura, seguido pela lenta e gradativa diminuição de sua temperatura, até que o ponto de solidificação seja atingido, quando o sistema atinge um estado de energia mínima (IGNACIO *et al.*, 2000). A SA pode ser utilizada na solução de diversos problemas de otimização combinatória. Muitos problemas de otimização florestal são de natureza combinatorial e, portanto, candidatos naturais à solução por esta técnica. Alguns exemplos incluem problemas com restrições de singularidade, adjacência, roteamento de veículos no transporte florestal, corte na indústria de móveis e de papel, dentre outros. Existem, contudo, poucos trabalhos reportando a utilização dessa técnica na solução desses problemas. LOCKWOOD e MOORE (1992), desenvolveram um algoritmo SA para resolver um problema de planejamento de colheita florestal com restrições de adjacência no Nordeste do Canadá. Ao invés de gerar apenas soluções viáveis para o problema, o algoritmo desenvolvido usou uma série de funções de penalidades que decresciam o valor da função objetivo para penalizar soluções inviáveis desencorajando a escolha dessas soluções. BOSTON e BETTINGER (1999), utilizaram a SA para resolver problemas de planejamento florestal com restrição de singularidade e adjacência e comparou-a com duas outras técnicas heurísticas: Programação Inteira Monte Carlo e Busca Tabu. Foram geradas quinhentas soluções para cada um dos quatro problemas que possuíam entre 3000 e 5000 variáveis inteiras 0-1. *Simulated Annealing* foi a técnica que encontrou o maior valor (cerca de 99%) para a função objetivo para três dos quatro problemas estudados.

Apesar de já existirem alguns estudos utilizando a metaheurística SA para resolver problemas de planejamento florestal, existem ainda algumas questões a

serem elucidadas para melhoria e consolidação dessa técnica na solução desses problemas. Tais melhorias podem incluir, por exemplo, novas estratégias de movimentos para melhorar a performance do método em problemas maiores, estudos adicionais sobre o efeito dos seus parâmetros na performance da heurística, critérios de parada mais informados, dentre outros. A inexistência de trabalhos desenvolvidos no Brasil é um outro fator que estimula novos estudos, contribuindo para disseminação desta técnica no meio florestal brasileiro. Assim sendo, idealizou-se este trabalho com os seguintes objetivos:

- Desenvolver e testar um algoritmo baseado na metaheurística *Simulated Annealing* para resolver problemas de planejamento florestal;
- Ilustrar uma aplicação da heurística em um problema de pequeno porte;
- Expandir a aplicação do algoritmo para problema maiores e mais complexos, através do desenvolvimento de um código computacional para esta finalidade.

2. MATERIAIS E MÉTODOS

2.1. O problema de planejamento florestal

Os problemas escolhidos para ilustrar o procedimento da heurística *Simulated Annealing* foram os mesmos propostos para ilustrar as aplicações do Algoritmo Genético e busca tabu, discutidos no item 2.1. do capítulo 4. O objetivo de cada problema consiste em maximizar o valor presente líquido sujeito às restrições de singularidade (adotar um único regime de manejo em cada talhão), produção mínima e produção máxima em cada período do horizonte de planejamento.

2.2. O Algoritmo *Simulated Annealing*

O algoritmo SA foi implementado no programa *SMAP-F* (Soluções Metaheurísticas Aplicadas ao Planejamento Florestal), codificado no ambiente *delphi* 5.0. Os testes com o algoritmo foram efetuados em um microcomputador

AMD K6II 500 MHZ, com memória RAM de 64 MB e disco rígido de 15GB. O esquema genérico do programa com o respectivo algoritmo pode ser resumido conforme fluxograma da Figura 1. Os detalhes desse algoritmo são descritos nos tópicos seguintes.

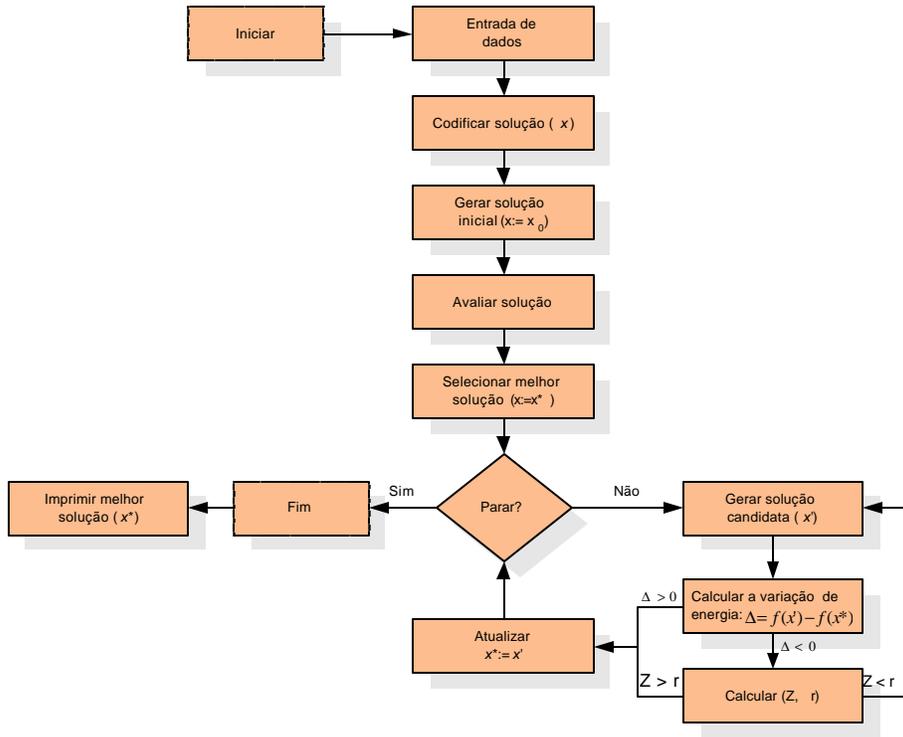


Figura 1 - *SMAP-F*: estrutura geral do programa para implementação da metaheurística *Simulated Annealing*.

2.2.1. Entrada de dados

A exemplo do Algoritmo Genético e Busca Tabu, a principal entrada de dados do algoritmo SA é constituída por um arquivo no formato texto gerado pelo sistema de suporte à decisão SisFlor, contendo o modelo de programação inteira representativo do problema em foco. Além do modelo para representar o problema, a solução de um problema utilizando a metaheurística SA requer a definição dos valores dos seus diferentes parâmetros. Este valores são definidos pelo usuário em um módulo específico, conforme discutido no item 2.3 desse capítulo.

2.2.2. Codificação da solução

A representação das alternativas de manejo para um dado problema é feita utilizando-se uma variável decisão X_{ij} para representar a alternativa de manejo j ($j=1,2,\dots,N$) assinalada à unidade de manejo i ($i=1,2,\dots,M$). Um vetor $V(x) = \{X_{11}, X_{12}, \dots, X_{ij}\}$, para representar essas variáveis decisão foi esquematizado conforme esquema da Figura 2.

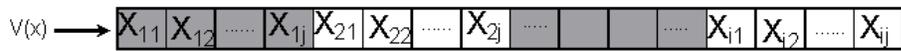


Figura 2 – Representação de um vetor com as variáveis de decisão do problema.

Uma solução (x) viável segundo a restrição de singularidade para o problema, formulado através do modelo de programação inteira com as variáveis de decisão na forma binária ($X_{ij} \in \{0,1\}$), avaliada segundo a função objetivo $f(x)$, foi representada de acordo com o esquema da Figura 3.



Figura 3 – Representação de uma solução para o problema com as variáveis de decisão na forma binária.

2.2.3. Solução inicial

A solução inicial (x_0) do algoritmo SA é gerada aleatoriamente. Com uma probabilidade uniformemente distribuída, uma variável de decisão X_{ij} é selecionada aleatoriamente em cada unidade de manejo para receber valor igual a 1 (variável básica). As demais variáveis são assinaladas com valores iguais a zero (variáveis não básicas), garantindo a viabilidade da restrição de singularidade, ou seja, adotar uma única opção de manejo em cada talhão.

2.2.4. Função de avaliação

A heurística SA utiliza o valor da função objetivo penalizada ($fp(x)$) como função de avaliação para guiar o processo de busca de novas soluções. A função objetivo $fp(x)$ é obtida através da modificação da função objetivo ($f(x)$) do

problema para aceitar soluções inviáveis, porém, aplicando-se a elas penalizações por violar restrições do problema desencorajando a sua utilização.

O valor de $f(x)$ de cada solução x é calculado utilizando a função objetivo do problema, dada por:

$$f(x) = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} \quad (1)$$

$f(x)$ = valor da função objetivo do problema para a solução x .

X_{ij} = variável de decisão que representa a j -ésima alternativa de manejo adotada na i -ésima unidade de manejo.

C_{ij} = é o valor presente líquido total (\$) de cada unidade de manejo i , colhida segundo a alternativa j ;

M = número total de unidades de manejo;

N = número total de alternativas de manejo na i -ésima unidade de manejo;

Conforme demonstrado no item 2.3.4.1. do capítulo 4, a expressão (1) pode ser alterada para penalizar movimentos que gerem soluções inviáveis, utilizando uma função objetivo penalizada, $f_p(x)$, dada por :

$$f_p(x) = \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} - vp \times VT \quad (2)$$

$f_p(x)$ = valor da função objetivo penalizada para a solução x .

VT ¹ = violação total (m³) das restrições de produção (mínima e máxima);

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

2.2.5. Movimentos

Os movimentos são estratégias utilizadas para geração de soluções candidatas. Na metaheurística SA apenas uma solução candidata (x') vizinha a x ($x' \in N(x)$) é avaliada em cada interação. Diversas estratégias de movimento podem ser definidas, sendo o seu desenho muitas vezes depende da natureza do problema e da criatividade do projetista. A estratégia utilizada neste trabalho utiliza um

¹ Obtido conforme demonstrado no item 2.3.4.1. do capítulo 4

mecanismo aleatório para geração desse movimento. O mecanismo de movimento adotado utiliza a seguinte estratégia: a partir de uma solução inicial (x_0) gerada aleatoriamente, a cada interação, um determinado número de unidades de manejo é escolhido aleatoriamente para ter uma variável X_{ij} adicionada (*Adc*) ($X_{ij}=1$). Ao parâmetro número de unidades de manejo escolhido para se fazer a troca de variáveis, denominou-se (*Ut*). A variável escolhida para ser removida da base (*Rem*) ($X_{ij}=0$) é aquela cujo valor na solução anterior que teve X_{ij} adicionada é igual a 1, satisfazendo dessa forma a restrição de singularidade. O Quadro 1 resume este procedimento com um movimento x' gerado através da adição da variável X_{11} e remoção da variável X_{12} a qual era uma variável básica na solução anterior (x) na unidade de manejo um. Neste caso, apenas uma unidade de manejo é escolhida para fazer a troca.

Quadro 1 - Geração de movimentos através do mecanismo de adicionar e remover aleatoriamente pares de variáveis de decisão (X_{ij}) em cada unidade de manejo

UM (i) VD	1			2			3			4			Movimentos	
	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	<i>Adc</i>	<i>Rem</i>
x	0	1	0	1	0	0	0	0	1	1	0	0	-	-
x'	1	0	0	1	0	0	0	0	1	1	0	0	X_{11}	X_{12}

A heurística SA aceita a nova solução x' se um dos seguintes critérios for satisfeito:

- valor da função objetivo $f(x') > f^*(x)$;
- valor da constante Z for maior do que um número aleatório (r) distribuído uniforme no intervalo entre 0 e 1, sendo:

$$Z = e^{\{f(x') - f^*(x)\}/T} \quad (3)$$

em que:

$f(x')$ = o valor da função objetivo da solução corrente;

$f^*(x)$ = valor da melhor função objetivo encontrada até o momento;

T = temperatura;

e = constante (2,718282); e

$r \in \text{uniforme}\{0,1\}$.

Uma vez o movimento x' tenha sido aceito, a solução x na iteração seguinte passa a ser $x := x'$. O algoritmo SA guarda também a melhor solução (x^*) encontrada até o momento, sendo a sua imagem $f^*(x)$ utilizada no cálculo de Z .

A unidade de manejo que recentemente tenha sido escolhida para ter uma troca efetuada pode permanecer por um dado período sem que nenhuma nova troca seja efetuada nesta unidade. Este período é definido como período sem troca (Pst) e refere-se ao número de iterações na qual uma dada unidade de manejo permanecerá sem que nenhuma troca seja efetuada em suas variáveis, o que equivale a proibir que a variável recentemente adicionada (Adc) seja removida (Rem) da base.

2.2.6. Temperatura

A temperatura (T) é um dos principais parâmetros da SA. Ela inicia com um valor alto, permitindo que movimentos de pior qualidade (ou que pioram a função objetivo) tenham maiores chances de serem escolhidos, sendo gradativamente reduzida até a convergência do algoritmo, o que equivale a diminuir as chances de escolha de um movimento de baixa qualidade, à medida que o algoritmo progride. Várias estratégias de redução da temperatura podem ser adotadas, e nenhum trabalho até o momento apresenta sugestões de qual o melhor esquema de redução dessa temperatura. Há um consenso na literatura especializada de que este parâmetro depende da natureza do problema, devendo ser definido por experimentação.

Três parâmetros são utilizados para definir a temperatura (T) em um dado instante: a temperatura inicial (T_0), a taxa de redução da temperatura (R_T) e o número de iterações na qual a temperatura permanece constante (K_{TC}). Um possível esquema poderia fornecer a temperatura num determinado instante como função potencial de T_0 , T_R e do número de interação k , da seguinte forma:

$$T_k = T_0 \left(1 - \frac{T_R}{100} \right)^k \quad (4)$$

em que:

T_k = temperatura na iteração k ;

T_0 = temperatura inicial;

T_R = taxa de redução da temperatura, expressa em porcentagem;

k = iteração.

2.2.7. Critério de parada

O algoritmo SA proposto utiliza o mesmo critério de parada definido para as heurísticas Algoritmo Genético e Busca Tabu desenvolvidos nos capítulos 4 e 5, respectivamente, ou seja: após encontrar a primeira solução com violação total nula ($VT=0$), a heurística SA executa um número de interações adicionais (Δk) para tentar melhorar a solução encontrada. Assim, a SA tenta encontrar várias soluções viáveis, escolhendo aquela de melhor valor para a função objetivo ($f^*(x)$). Para interromper a corrida da SA após um número de iterações sem encontrar uma solução viável para o problema, foi definido um número máximo de iterações (k_{max}).

2.3. Escolha dos parâmetros da SA

Os parâmetros da metaheurística SA avaliados neste trabalho foram a temperatura inicial (T_0), a taxa de redução da temperatura (T_R) e o número de iterações com a temperatura mantida constante (K_{TC}). Além desses parâmetros básicos, a metaheurística SA desenvolvida utiliza outros parâmetros com diferentes finalidades. Na estratégia de movimentos dessa metaheurística foram utilizados os parâmetros número unidades de manejo alteradas (Ut) e período sem troca (Pst), com respectivas finalidades de definir o número de unidades de manejo escolhidas para se fazer a troca de variáveis e o número de iterações para a qual a unidade de manejo recentemente trocada permaneça sem que nenhuma troca seja efetuada. Outros parâmetros da SA foram: o valor da penalização (vp) utilizado nos cálculos das penalidades; o número de interações adicionais (Δk) para melhorar a qualidade

da solução utilizada no critério de parada; e o número máximo de iterações permitidas (k_{max}).

A Figura 4 mostra uma janela da rotina computacional, com opções para escolha dos parâmetros utilizados em uma dada corrida da SA.

Os valores dos parâmetros utilizados neste trabalho são apresentados no Quadro 2. As combinações entre os níveis dos diferentes parâmetros gerou um ensaio com 54 diferentes combinações, representando cada uma delas um cenário ou tratamento diferente. Cada tratamento foi repetido 5 vezes, resultando em 270 observações para cada problema. Considerando os quatro problemas utilizados no teste, gerou-se um ensaio com 1080 observações.

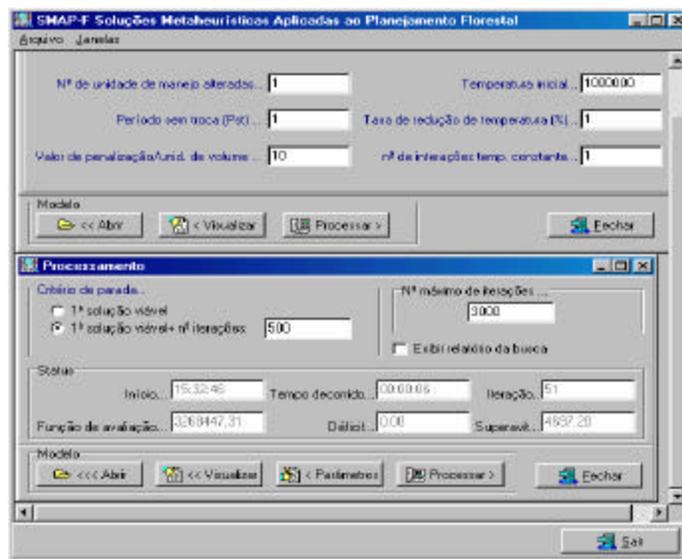


Figura 4 – SMAP-F: interface computacional para definição dos parâmetros da SA.

Quadro 2 - Resumo dos parâmetros utilizados pela metaheurística *Simulated Annealing*

Parâmetros	Níveis	Valores dos parâmetros		
Iterações com temperatura constante (K_{TC})	3	1	50	100
Temperatura inicial (T_0)	3	10^6	10^7	10^8
Taxa de redução da temperatura (T_R) - %	3	0,5	1,0	5,0
Unidades de manejo trocadas (U_t)	1	1		
Período sem troca (Pst)	2	0	M/2	
Valor da penalização (vp) – $\$/m^3$	1	10		
Número máximo de iterações (k_{max})	1	3000		

Iterações adicionais (Δk)	1	500		
-------------------------------------	---	-----	--	--

em que: M = número de unidades de manejo do problema

2.4. Avaliação do desempenho da SA

A performance da SA foi avaliada através da sua eficácia, calculada pela razão, expressa em porcentagem, entre o melhor valor da função objetivo obtida em uma corrida da SA e o ótimo matemático, obtido pelo algoritmo exato *branch and bound*, assim:

$$Ef = \frac{f_{SA}}{f_o} \times 100 \quad (5)$$

em que:

Ef = eficácia (%);

f_{SA} = valor da melhor solução obtida em uma corrida da SA (\$);

f_o = valor da solução ótima obtida pelo algoritmo *branch and bound* (\$);

A eficiência da SA foi avaliada utilizando-se apenas medidas de tempo de processamento do algoritmo, o que permite somente comparações relativas entre os métodos de solução, processadas utilizando-se os mesmos recursos computacionais.

2.5. Análise estatística dos resultados

As comparações entre os valores dos diferentes parâmetros foram efetuadas utilizando-se o procedimento estatístico proposto por LEITE e OLIVEIRA (2000) para comparar métodos analíticos. O procedimento proposto por estes autores, denominado teste L&O, utiliza uma metodologia que combina o teste F de GRAYBILL (F(H0)), o teste t para o erro médio ($\bar{\epsilon}$) e o coeficiente de correlação linear (r_{yij}) entre as variáveis. Além do teste L&O utilizou-se a estatística descritiva para auxiliar a análise dos resultados.

3. RESULTADOS E DISCUSSÃO

3.1 Ilustração de uma corrida da heurística SA

No Quadro 3 é apresentado uma corrida da heurística *Simulated Annealing* aplicada ao problema um, cujo objetivo foi a maximização do valor presente líquido sujeito às restrições de singularidade e produção mínima. Este pequeno problema possui 12 variáveis de decisão com 4 restrições de singularidade e 4 restrições de produção mínima. Os parâmetros da SA escolhidos para ilustrar esta corrida empregou uma temperatura inicial (T_0) igual a 50000 aplicando uma taxa de redução (T_R) de 1% a cada iteração ($K_{TC} = 1$). A temperatura inicia com um valor mais alto (50000) sendo gradativamente reduzida a cada iteração até o final da corrida.

O algoritmo inicia o processo de busca gerando uma solução inicial (x_0) aleatoriamente, ou seja, escolhe-se em cada unidade de manejo uma única variável para ser adicionada com valor igual a 1, garantido a não violação da restrição de singularidade. A solução corrente (x) passa a ser então x_0 ($x := x_0$). Após a solução ter sido gerada calculam-se os valores da função objetivo penalizada, $fp(x)$, e não penalizada ($f(x)$), além da violação total (VT) que cada solução provoca na restrição de produção mínima. Algoritmo continua o processo de busca gerando a cada iteração (k) uma solução candidata (x'), na vizinhança da solução corrente (x). Na primeira iteração ($k=1$) a solução candidata (x') é a solução x_1 , gerada a partir de x_0 através da adição (*Adc*) da variável X_{41} e remoção (*Rem*) da variável X_{43} . A escolha da variável de decisão para ser adicionada é feita aleatoriamente. Após a nova solução ter sido gerada, testes para avaliar se o movimento é permitido são efetuados. O primeiro teste é comparar a função objetivo penalizada da solução candidata ($fp(x')$), utilizada como critério para seleção de movimentos, com a melhor solução encontrada até o momento (na iteração um, o valor da melhor solução encontrada até o momento é $fp(x_0)$, ou seja, 178000). Deste modo, na primeira iteração compara-se o valor de x_1 (198.000) com valor (178000). Uma vez que o valor da função objetivo penalizada da solução candidata (x_1) é maior do que o valor da melhor solução encontrada até o momento, a solução x_1 é aceita e passa a ser a nova solução na iteração seguinte ($k=2$). Na iteração dois, a vizinhança de x_1 é representada pela solução candidata x_2 , com um valor da respectiva função objetivo

igual a 186.000, portanto, menor do que o valor da melhor solução encontrada até o momento (198.000). Neste caso, um segundo critério é avaliado para decidir se o movimento será aceito. Este teste utiliza a variável Z como critério de escolha, comparando-a com um valor aleatório (r), uniformemente distribuído entre 0 e 1. O movimento é aceito toda vez que Z for superior a r . No exemplo em pauta, para a solução x_2 , tem-se Z igual 0,78 e r igual a 0,85, o que resulta na rejeição da solução x_2 . Prosseguindo na busca, na iteração três, uma nova solução candidata (x_3) é gerada para a solução x_1 . A solução candidata x_3 é aceita já que o valor da sua função objetivo penalizada é maior do que o melhor valor encontrado até o momento. Na iteração quatro, a solução candidata x_4 , apesar de apresentar um valor da função objetivo penalizada (206.000) inferior ao valor da melhor solução encontrada até o momento (218000), foi selecionada, uma vez que, o valor de Z (0,77) é superior ao valor de r (0,32).

Na iteração oito, é encontrada uma solução com violação total nula ($VT = 0$). Esta solução é também a solução ótima para o problema. A rápida convergência para a solução ótima deve-se, obviamente, ao pequeno número de variáveis do problema e a sua relativa simplicidade.

Quadro 3 – Ilustração de uma corrida da heurística *Simulated Annealing* para o problema um, considerando uma temperatura inicial (T_0) igual a 50000, taxa de redução da temperatura (T_R) igual a 1% e iterações com temperatura constante (K_{TC}) igual a um

		Variáveis de Decisão																						
x	x'	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}	Adc	Rem	$f(x)$	VT	$fp(x)$	Z	r	$fp^*(x)$	T	k	
x_0	-	1	0	0	0	0	1	1	0	0	0	0	1	-	-	378000	20000	178000	-	-	-	50000	-	
x_0	x_1	1	0	0	0	0	1	1	0	0	1	0	0	X_{41}	X_{43}	298000	10000	198000	-	-	178000	49500	1	
x_1	x_2	1	0	0	0	1	0	1	0	0	1	0	0	X_{22}	X_{23}	286000	10000	186000	0.78	0.85	198000	48515	2	
x_1	x_3	0	1	0	0	0	1	1	0	0	1	0	0	X_{12}	X_{11}	318000	10000	218000	-	-	198000	47074	3	
x_3	x_4	0	1	0	0	1	0	1	0	0	1	0	0	X_{22}	X_{23}	306000	10000	206000	0.77	0.32	218000	45219	4	
x_4	x_5	0	0	1	0	1	0	1	0	0	1	0	0	X_{13}	X_{12}	326000	10000	226000	-	-	218000	43003	5	
x_5	x_6	0	0	1	0	1	0	0	1	0	1	0	0	X_{32}	X_{31}	341000	20000	141000	0.12	0.55	226000	40486	6	
x_5	x_7	0	0	1	0	1	0	1	0	0	0	1	0	X_{42}	X_{41}	366000	20000	166000	0.20	0.74	226000	37736	7	
x_5	x_8	0	0	1	0	0	1	1	0	0	1	0	0	X_{23}	X_{22}	338000	0	338000	-	-	226000	34821	8	
x_8	-	0	1	0	0	0	1	1	0	0	1	0	0	X_{12}	X_{13}	318000	10000	218000	0.02	0.03	338000	31809	9	

3.2. Avaliação dos parâmetros e medidas de performance da SA

Os resultados do teste L&O para comparar os valores da temperatura inicial, o número de iterações com temperatura constante, taxa de redução da temperatura e período sem troca são mostrados no Quadro 4. Houve diferenças estatísticas significativas entre alguns níveis ou categorias destes parâmetros para os problemas analisados a um nível de significância de 1% de probabilidade.

Não houve efeito estatisticamente significativo, pelo teste de L&O a um nível de significância de 1%, do parâmetro “período sem troca” (*Pst*) na eficácia da SA (Quadro 4). Apesar de não ter sido estatisticamente significativo, este parâmetro representou uma das inovações da metaheurística SA desenvolvida neste trabalho em relação aos principais modelos reportado na literatura. Semelhante ao parâmetro permanência, utilizado pela Busca Tabu, o objetivo principal desse parâmetro é proibir a execução de movimentos de adição e remoção de variáveis em uma unidade de manejo que tenha sido recentemente alterada (i.e., tenha ocorrido um movimento de trocas de variáveis através do movimento de adição e remoção na respectiva unidade de manejo). O fato de proibir movimentos em algumas unidades de manejo direciona a busca para novas regiões do espaço de solução e funciona como um mecanismo de diversificação da busca.

O comportamento da temperatura ao longo de uma corrida do algoritmo SA, considerando taxas de redução da temperatura de 0,5, 1,0 e 5,0% é apresentado na Figura 5a. Neste exemplo, os valores da temperatura inicial foi igual a 1.000.000, sendo reduzida a cada iteração, ou seja, o valor do parâmetro número de iterações com temperatura constante é igual a um ($K_{TC}=1$). A temperatura é o principal parâmetro da SA podendo afetar a sua performance. Temperaturas altas aumentam a probabilidade de um movimento de pior qualidade (valor inferior para função objetivo) ser aceito, o que confere maior diversidade às soluções no início da corrida. A diminuição na temperatura tem um efeito contrário, diminuindo as chances dos movimentos de pior qualidade serem selecionados e, conseqüentemente, reduz a diversidade das soluções candidatas, forçando o

algoritmo a convergir para um ótimo local ou global. Valores muito altos de temperaturas são importantes para manter a diversidade do espaço de solução, mas pode dificultar o algoritmo a convergir para uma solução boa. Valores muito baixos, ao contrário, forçam a convergência rápida do algoritmo, mas podem levá-lo a estacionar em ótimos locais. Portanto, o equilíbrio desses fatores pode ser decisivo para o sucesso do algoritmo. Não existe, ainda, na literatura especializada recomendações seguras a respeito do melhor esquema, tanto para temperaturas iniciais quanto para o programa ideal de redução. Tais escolhas são ainda baseadas na experimentação. Os valores ideais são aqueles que garantem um equilíbrio entre a diversificação e a intensificação da busca. Uma contribuição recente em relação a adoção de procedimentos mais sólidos para escolha da temperatura inicial pode ser encontrada em YOUSSEF et al. (2001).

Não houve diferenças estatísticas significativas das temperaturas iniciais adotadas na eficácia da SA, pelo teste de L&O a um nível de 1% de probabilidade. Na Figura 5b é apresentada a medida de eficácia do algoritmo SA para os três diferentes valores temperaturas iniciais adotados.

Em relação às taxas de redução da temperaturas empregados, a exceção dos valores de 0,5 e 1,0, as demais comparações diferiram entre si pelo teste de L&O, a um nível de 1% de probabilidade (Quadro 4). As Figura 5c e 5d ressaltam os efeitos desses parâmetros, onde na Figura 5d é mostrada a respectiva frequência de valores por classe de eficácia.

Um outro parâmetro da SA avaliado neste trabalho foi o número de iterações com temperatura mantida constante (K_{TC}). Apenas as comparações entre os valores iguais a 50 e 100 não diferiram entre si pelo teste de L&O a um nível de 1% de probabilidade (Quadro 4). A Figuras 5e resalta que o valor de K_{TC} igual a um ($K_{TC}=1$) propiciou uma melhor eficácia da SA, já que maiores valores de eficácia são produzidos com este valor (Figura 5e). Este comportamento é evidenciado na Figura 5f, a qual mostra que o valor de K_{TC} igual a um apresentou maiores frequências nas maiores classes de eficácia quando comparado aos valores 50 e 100. Estes resultados mostram um aspecto interessante da heurística SA, uma vez que, o simples fato de mudar a estratégia de redução da temperatura ao longo da busca

pode afetar o desempenho do algoritmo. BOSTON e BETTINGER (1999) avaliando o efeito desses parâmetros para a heurística SA aplicada a problemas de planejamento florestal com restrições de singularidade, produção mínima e adjacência, com tamanho entre 3000 e 5000 variáveis de decisão, também encontraram resultados diferenciados. Avaliando o efeito de temperaturas iniciais iguais a 500000, 900000 e 1500000, cada uma delas mantidas constantes durante três períodos distintos de 100, 200 e 300 iterações, com taxas de redução após cada um desses períodos de 1%, 2,5%, e 5%, os autores concluíram que temperaturas mais altas (900000 e 1500000) propiciaram melhores performance para a heurística e não apresentaram diferenças significativas entre si. Os melhores intervalos para temperaturas mantidas constantes foram 200 e 300 iterações sem diferenças entre si. A melhor taxa de redução foi de 1%. Infelizmente não há uma regra simples para determinar a melhor combinação dos diferentes parâmetros da metaheurística SA para todos os problemas. A melhor configuração para um dado problema é ainda dependente da experimentação.

A eficácia da heurística SA oscilou entre os valores de 80 e 100% para os diferentes parâmetros avaliados, conforme pode ser observado nos gráficos da Figura 5 (b, c, e). Entretanto, quando considerou-se apenas o valor de iterações com a temperatura mantida constante igual a um ($K_{TC}=1$), a metaheurística SA apresentou um melhor desempenho com uma eficácia média igual a 95,36%, valor mínimo igual a 83,66, valor máximo igual a 100% e coeficiente de variação igual a 3,18%.

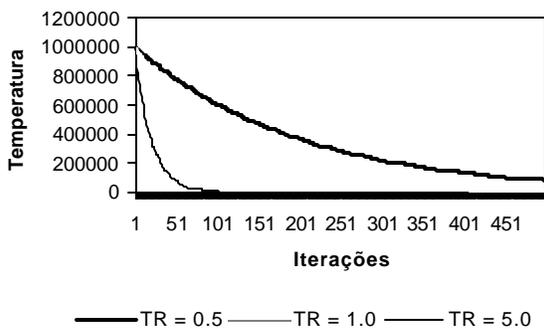
Quadro 4 - Teste de L&O² para comparar os efeitos dos parâmetros da metaheurística *Simulated Annealing*

Temperatura inicial (T_0)		F(H0)	t	r
10^5	10^6	n.s	n.s	sim
10^5	10^7	n.s	n.s	sim
10^6	10^7	n.s	n.s	sim
Iterações com a temperatura constante (K_{TC})		F(H0)	t	r
1	50	n.s	*	sim

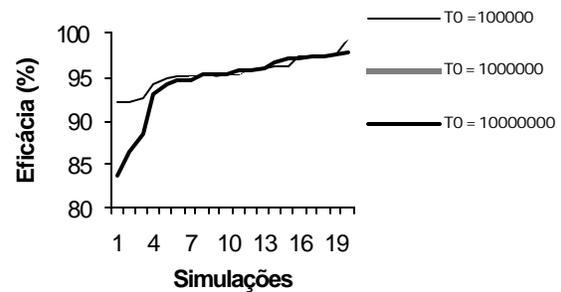
² Regra decisória obtida conforme item 2.3.12. do capítulo 4.

1	100	n.s	*	sim
50	100	n.s	n.s	sim
Taxa de redução da temperatura (T_R)		F(H0)	t	r
0,5	1,0	n.s	n.s	sim
0,5	5,0	n.s	*	sim
1,0	5,0	n.s	*	sim
Período sem troca (Pst)		F(H0)	t	r
Com	Sem	n.s	n.s	sim

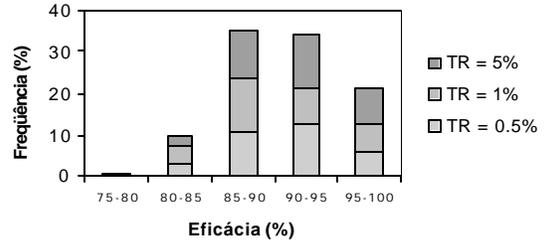
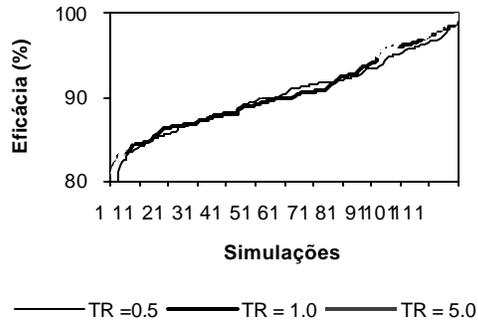
A eficiência da SA não foi avaliada rigorosamente nesse trabalho, o que requereria análise de algoritmos mais sofisticadas como complexidade de tempo e de espaço. Entretanto, os resultados computacionais preliminares já mostraram vantagens da SA em relação ao algoritmo exato *branch and bound*. Na Figura 6 é apresentado o tempo de processamento da SA e do algoritmo *branch and bound*. Pode-se observar que as diferenças entre o tempo de processamento dos dois algoritmos aumenta à medida que cresce o número de variáveis de problema. Para o problema com 423 variáveis, o algoritmo *branch and bound* gastou cerca de dez vezes mais tempo do que a SA para encontrar a solução do problema. Cabe ressaltar que esta vantagem pode ser melhorada através de aperfeiçoamento na estrutura do algoritmo desenvolvido e não foi ainda explorada neste trabalho.



(a)

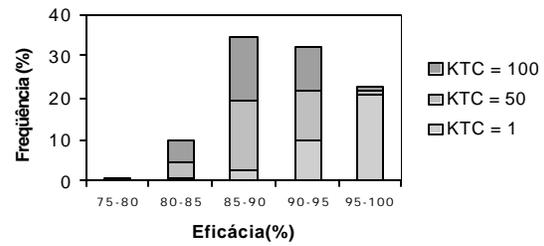
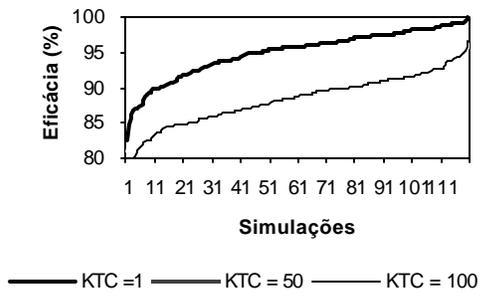


(b)



(c)

(d)



(e)

(f)

Figura 5 – Variação da temperatura (a), medida de eficácia para diferentes valores de temperatura inicial (b), medida de eficácia e respectivas frequência para diferentes taxas de redução da temperatura (c, d) e número de iterações com temperaturas mantidas constantes (e, f) para o algoritmo *Simulated Annealing*.

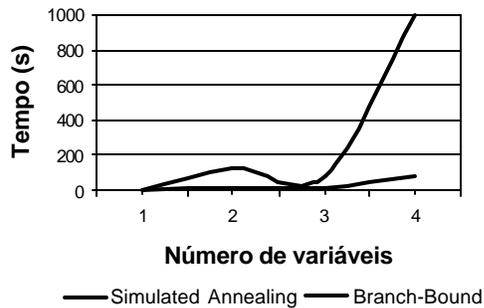


Figura 6 – Comparações do tempo de solução da metaheurística *Simulated Annealing* e o algoritmo *branch and bound*.

4. CONCLUSÕES

O presente estudo utilizou a metaheurística *Simulated Annealing* para resolver problemas de planejamento florestal com restrições de inteireza. Foram utilizados cinco problemas testes contendo entre 12 e 423 variáveis de decisão, onde o objetivo de cada problema foi a maximização da receita líquida, sujeita às restrições de singularidade e produção mínima e máxima.

O problema 1, com 12 variáveis de decisão, com restrição de singularidade e produção mínima, foi utilizado para ilustrar uma corrida da metaheurística SA. Os demais problemas foram utilizados para avaliar a performance da SA em função das mudanças de alguns de seus parâmetros. A performance foi avaliada pela medida de eficácia, expressa a relação entre o valor da função objetivo encontrada pela SA e o ótimo matemático obtido pelo algoritmo exato *branch and bound*. Com base nos resultados pode-se concluir que:

A eficácia da SA foi estatisticamente influenciada pelos valores ou categorias dos diferentes parâmetros empregados, avaliada pelo teste L&O ao nível de 1% de probabilidade.

A estratégia de redução da temperatura forneceu melhores valores de eficácia quando o número de iterações mantidas constante foi igual a um, comparado aos valores 50 e 100.

Utilizando os melhores parâmetros da SA é possível obter soluções próximas do ótimo matemático (cerca de 95% do valor ótimo matemático)

O tempo de processamento do algoritmo SA é muito inferior (neste estudo, cerca de dez vezes) ao principal algoritmo exato de solução (*branch and bound*).

Baseado nos resultados deste trabalho pôde-se concluir que a metaheurística SA é flexível por comportar restrições de diversas naturezas sem a necessidade de alterar a estrutura do algoritmo adotado.

A metaheurística SA pode ser usada para resolver uma variedade de problemas de planejamento florestal, entre eles, aqueles que envolvem restrições de adjacência, roteamento de veículos para o transporte florestal, corte na indústria, seccionamento de toras, dentre outros.

Os objetivos do trabalho foram alcançados, mas a aplicação da SA em problemas reais é dependente de melhorias na capacidade do algoritmo em resolver problemas de grande porte, típicos do setor florestal.

A melhor performance da SA é dependente da escolha de uma boa configuração de seus parâmetros. A inexistência de conclusões seguras para a maioria dos problemas de planejamento florestal, em especial, os problemas maiores e, ou mais complexos, sugerem que essa escolha deve ser baseada na experimentação.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- BOSTON, K.; BETTINGER, P. An Analysis of Monte Carlo Integer Programming, Simulated Annealing, and Tabu Search Heuristics for Solving Spatial Harvest Scheduling Problems.** *Forest Science, Washington*, **45(2):292–301, 1999.**
- CERNY, V. Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm.** *Journal of Optimization Theory and Applications*, **45(1): 41-51, 1985.**
- IGNACIO, A.A.V.; FERREIRA FILHO, V.J.M.; GALVÃO, R.D. Métodos heurísticos num entorno paralelo. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. *Anais*. Viçosa: UFV, 2000. p769-788.
- KIRKPATRICK Jr., S., GELATT, C., VECCHI, M. Optimization by simulated annealing.** *Decision Science* **220(4598), 498-516, 1983.**
- LEITE, H.G. & OLIVEIRA, F.H.T. Statistical Procedure to Test the Identity of Analytical Methods. *Comm. Soil Sci. Plant Anal.*, 2001, 16p. (no prelo)
- LOCKWOOD, C., MOORE, T. Harvest scheduling with spatial constraints: a simulated annealing approach. *Can. J. For. Res.* **23, 468-478, 1992.**

YOUSSEF, H.; SAIT, S.M.; ADICHE, H. Evolutionary algorithms, simulated annealing and tabu search: a comparative study. **Engineering Applications of Artificial Intelligence**. 14: 167-181, 2001.

RESUMO E CONCLUSÕES

Este trabalho avaliou a contribuição dos modelos, dados, tecnologias e recursos humanos no processo de tomada de decisão em gerenciamento de recursos florestais. A importância da tecnologia foi ressaltada através do desenvolvimento e apresentação de uma primeira versão de um sistema de suporte à decisão para auxiliar o manejador florestal. Foram avaliadas também as limitações de alguns modelos e algoritmos utilizados na formulação e solução de problemas neste setor. A contribuição das metaheurísticas para resolver as limitações dos principais algoritmos exatos de solução foi ressaltada e, em seguida, proposto o desenvolvimento um modelo computacional constituído pelas metaheurísticas Algoritmo Genético (AG), Busca Tabu (BT) e *Simulated Annealing* (SA). O trabalho foi desenvolvido na forma de capítulos permitindo-se tirar as conclusões apontadas a seguir:

- **O gerenciamento florestal é uma atividade complexa e o manejador florestal não pode abrir mão do uso de abordagens científicas para auxiliá-lo. Modelos, dados, tecnologias e recursos humanos são partes integrantes dessas abordagens.**
- Os modelos de suporte à decisão no gerenciamento florestal são muito variados e incluem, por exemplo, os modelos de prognose da produção florestal, modelos econômicos, modelos de Pesquisa Operacional e heurísticos. O sucesso desses modelos dependem da utilização de dados com qualidade e níveis de detalhes adequados e dependem de investigações em diferentes disciplinas.
- As tecnologias de informação são componentes indispensáveis ao processo decisório, e têm contribuído para facilitar as aplicações dos modelos e uso de

dados principalmente em problemas de grande escala, gerando informações ao tomador de decisão com rapidez e na frequência desejada.

- Embora sejam indispensáveis o uso das tecnologias de informação para suporte à decisão no planejamento florestal, o conhecimento acumulado dos manejadores experientes não podem ser desprezados.
- Muitas barreiras enfrentadas no tratamento de problemas de planejamento florestal não são devidas à falta de tecnologias ou de dados para resolvê-los, e sim, pela carência de pessoal qualificado para formular e resolver problemas utilizando tais recursos.
- Dentre as tecnologias de informação, os sistemas de suporte à decisão ocupam um lugar de destaque na formulação e solução de problemas variados. Com capacidade para utilizar dados, modelos e conhecimento de especialista na solução de problemas, auxiliam o manejador florestal em diversas etapas do processo decisório.
- Apesar das potencialidades inquestionáveis desses sistemas, seu uso em muitas empresas florestais é ainda bastante incipiente. A escassez de sistemas adequados à realidade florestal de muitas empresas e a falta de reconhecimento por parte de alguns manejadores das suas potencialidades, são alguns fatores que têm contribuído para a sua pouca difusão.
- O sistema SisFlor (sistema de suporte à decisão otimizada para o planejamento florestal), desenvolvido neste trabalho, além da facilidade de uso, se destaca pela grande flexibilidade do seu módulo gerador de alternativas de manejo, contribuindo para a formulação de importantes questões do gerenciamento de florestas equiâneas.
- Apesar dos avanços, algumas melhorias nas interfaces do SisFlor, tais como: entrada de dados e relatórios, integração com outros sistemas, em especial com os sistemas de cadastro, inventário e sistemas de informação geográfica, necessitam ser efetuadas para facilitar a sua utilização em problemas de grande escala.
- Os modelos de Pesquisa Operacional utilizados pelo SisFlor na formulação de problemas são os de Programação Linear e Programação Inteira, solucionados

através do algoritmo *simplex* e *branch and bound*, respectivamente, disponibilizados no pacote comercial LINDO.

- O algoritmo *simplex* é altamente eficiente em problemas de grande porte, mas apresenta restrições de aplicações em problemas que requerem o uso de variáveis inteiras.
- O algoritmo *branch and bound* é eficiente somente em problemas de pequeno porte, o que não é o caso da maioria dos problemas reais de planejamento florestal que, em geral, envolvem centenas a milhares de variáveis de decisão.
- Técnicas heurísticas para a solução de problemas com variáveis inteiras tem sido desenvolvidas com muito sucesso em diversas áreas. As heurísticas tidas como mais promissoras neste campo pertencem as chamadas metaheurísticas, dentre elas, Busca Tabu (BT), *Simulated Annealing* (SA) e Algoritmos Genéticos (AG).
- Neste trabalho, as metaheurísticas AG, BT e SA foram implementadas na rotina computacional *SMAP-F* (Soluções Metaheurísticas Aplicadas ao Planejamento Florestal).
- Os efeitos dos diferentes parâmetros dessas metaheurísticas foram avaliados segundo medida de eficácia, em relação ao ótimo matemático obtido pelo algoritmo *branch and bound* em cinco problemas com restrições de singularidade, produção mínima e produção máxima.
- A eficiência, medida pelo tempo de processamento para obtenção da solução através da BT, AG e SA foram cerca de duas, cinco e dez vezes superior ao algoritmo *branch and bound*, respectivamente.
- As três metaheurísticas apresentaram-se como abordagens atrativas para solução de problemas de Programação Inteira, muito comum no gerenciamento florestal, problemas esses de difícil solução ou até mesmo impossíveis de serem solucionados pelos algoritmos exatos da atualidade.
- Aplicabilidade universal das metaheurísticas estudadas, propiciada pela flexibilidade desses algoritmos em resolver problemas de diferentes domínios, permite inferir que esta técnica pode ser utilizada para resolver uma variedade de outros importantes problemas de planejamento florestal, entre eles, aqueles que

envolvem restrições de adjacência, roteamento de veículos para o transporte florestal, corte na indústria, seccionamento de toras etc.

- A utilização das metaheurísticas apresentadas neste trabalho está apenas iniciando no setor florestal, portanto, muitos de seus aspectos necessitam ser melhor investigados. Outras estratégias de movimentos, novas funções de avaliação, outros critérios de interrupção e uma melhor investigação do comportamento dos seus parâmetros, principalmente em problemas maiores e mais complexos, são alguns desses aspectos.
- Os objetivos propostos neste trabalho, em relação ao desenvolvimento e aplicação das metaheurísticas, foram alcançados. Entretanto, as pesquisas para melhorar a capacidade dos algoritmos em resolver problemas maiores necessitam ser continuadas para viabilizar suas aplicações em problemas reais.