

BRUNA BÁRBARA SANTOS

**PROCEDIMENTOS HÍBRIDOS NA EXPLORAÇÃO DO ESPAÇO DE
SOLUÇÕES DE PROBLEMAS DE NATUREZA COMBINATÓRIA: UMA
APLICAÇÃO AO PLANEJAMENTO FLORESTAL**

Dissertação apresentada à
Universidade Federal de Viçosa,
como parte das exigências do
Programa de Pós-Graduação em
Ciência da Computação, para
obtenção do título de *Magister
Scientiae*.

**VIÇOSA
MINAS GERAIS – BRASIL
2007**

**Ficha catalográfica preparada pela Seção de Catalogação e
Classificação da Biblioteca Central da UFV**

T

S237p
2007

Santos, Bruna Bárbara, 1982-

Procedimentos hídricos na exploração do espaço de
soluções de problemas de natureza combinatória: uma
aplicação ao planejamento florestal / Bruna Bárbara
Santos. - Viçosa, MG , 2007.
x, 50f. : il. ; 29cm.

Orientador: Heleno do Nascimento Santos.
Dissertação (mestrado) - Universidade Federal de
Viçosa.

Referências bibliográficas: f. 46-50.

1. Pesquisa operacional. 2. Solução de problemas.
3. Algoritmos. 4. Processo decisório. 5. Engenharia
florestal. I. Universidade Federal de Viçosa. II. Título.

CDD 22.ed. 003

BRUNA BÁRBARA SANTOS

**PROCEDIMENTOS HÍBRIDOS NA EXPLORAÇÃO DO ESPAÇO DE
SOLUÇÕES DE PROBLEMAS DE NATUREZA COMBINATÓRIA: UMA
APLICAÇÃO AO PLANEJAMENTO FLORESTAL**

Dissertação apresentada à
Universidade Federal de Viçosa,
como parte das exigências do
Programa de Pós-Graduação em
Ciência da Computação, para
obtenção do título de *Magister
Scientiae*.

APROVADA: 28 de Fevereiro de 2007

Prof. Luiz Aurélio Raggi
(Co-Orientador)

Prof. Vladimir Oliveira Di Iorio
(Co-Orientador)

Prof. Mauro Nacif Rocha

Prof. Gilson Fernandes da Silva

Prof. Heleno do Nascimento Santos
(Orientador)

DEDICATÓRIA

Aos meus pais Marcos e Wandira, aos meus irmãos Maíra e Thiago e ao noivo Andrey.

AGRADECIMENTOS

Primeiramente, a Deus pela vida.

À Universidade Federal de Viçosa, através do Departamento de Informática, pela oportunidade de realizar este curso.

Ao Prof. Heleno do Nascimento Santos, pela orientação e apoio indispensáveis em todas as etapas deste trabalho.

À CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), pelo apoio financeiro.

Aos meus pais Marcos e Wandira, pelo amor incondicional.

Aos meus irmãos Maíra e Thiago, e a todos os demais familiares, que estiveram sempre torcendo pelo meu sucesso.

Ao meu noivo Andrey, pelo amor, carinho e força nos momentos mais difíceis.

Aos meus amigos de curso pela agradável convivência e pela ajuda nos estudos.

Aos professores e funcionários do Departamento de Informática da Universidade Federal de Viçosa pela total colaboração aos meus estudos.

A todos aqueles que participaram da realização deste trabalho, direta ou indiretamente.

BIOGRAFIA

Bruna Bárbara Santos, filha de Marcos Alves dos Santos e Wandira Vasconcelos Bárbara Santos, nasceu no dia 9 de janeiro de 1982 na cidade Passos – MG.

Em dezembro de 1999 concluiu o ensino médio no Colégio Imaculada Conceição.

Em agosto de 2000, ingressou na Universidade Estadual de Minas Gerais através da Fundação de Ensino Superior de Passos, obtendo o título de Tecnóloga em Informática em dezembro de 2002.

Em fevereiro de 2003, ingressou na Universidade de Franca, obtendo o título de Especialista em Sistemas de Informação em julho de 2006.

Em março de 2004, iniciou o curso de mestrado no Departamento de Informática da Universidade Federal de Viçosa.

É docente da Fundação de Ensino Superior de Passos, desde fevereiro de 2006.

SUMÁRIO

LISTA DE SIGLAS	VI
LISTA DE FIGURAS	VII
RESUMO	VIII
ABSTRACT.....	IX
1. INTRODUÇÃO	1
1.1. OBJETIVOS	3
1.2. ORGANIZAÇÃO DO TRABALHO	4
2. REVISÃO BIBLIOGRÁFICA.....	6
2.1. PLANEJAMENTO DA PRODUÇÃO FLORESTAL	6
2.2. PROGRAMAÇÃO POR RESTRIÇÕES.....	15
2.3. METAHEURÍSTICAS.....	22
2.3.1. <i>Método de Trajetória: Simulated Annealing</i>	24
2.3.2. <i>Método Baseado em População: Algoritmo Genético</i>	25
2.3.3. <i>Metaheurísticas Paralelas</i>	25
3. MATERIAIS E MÉTODOS	28
3.1. HIBRIDIZAÇÃO USANDO PROGRAMAÇÃO POR RESTRIÇÕES E METAHEURÍSTICA.	30
3.2. HIBRIDIZAÇÃO DA METAHEURÍSTICA ALGORITMO GENÉTICO.	38
4. RESULTADOS E DISCUSSÃO	39
4.1. HIBRIDIZAÇÃO USANDO PROGRAMAÇÃO POR RESTRIÇÕES E METAHEURÍSTICA.	39
4.2. HIBRIDIZAÇÃO DA METAHEURÍSTICA ALGORITMO GENÉTICO.	42
5. CONCLUSÃO	44
REFERÊNCIAS BIBLIOGRÁFICAS.....	46

LISTA DE SIGLAS

AG	Algoritmos Genéticos
BT	Busca Tabu
CACP	<i>Computer-Aided Constraint Programming</i>
CLPFD	<i>Constraint Logic Programming for Finite Domain</i>
CSP	<i>Constraint Satisfaction Problem</i>
EaCL	<i>Easy abstract Constraint Optimization Programming Language</i>
GFC	<i>General Forward Checking</i>
GLS	<i>Guided Local Search</i>
IA	Inteligência Artificial
PL	Programação Linear
PPV	Problema de Programação de Viagens
PR	Programação por Restrições
PSA	<i>Pareto Simulated Annealing</i>
SA	Simulated Annealing
SPEA	<i>Strength Pareto Evolutionary Algorithm</i>
SSD	Sistemas de Suporte à Decisão

LISTA DE FIGURAS

<i>Figura 1</i> – Mapa da Austrália.....	17
<i>Figura 2</i> – Parte de uma árvore de busca.....	18
<i>Figura 3</i> – Integração entre o <i>Delphi</i> e o <i>GNU Prolog</i>	20
<i>Figura 4</i> – O esqueleto de um problema em <i>EaCL</i>	32
<i>Figura 5</i> – Especificação do CSP para o problema da pintura do mapa.....	33
<i>Figura 6</i> – Problema do caixeiro-viajante descrito na linguagem de programação <i>EaCL</i>	35
<i>Figura 7</i> – Visão da ferramenta <i>ZDC</i> , em sua interface gráfica.....	37
<i>Figura 8</i> – Tela da ferramenta <i>ZDC</i> gerando as prescrições do problema teste proposto.....	40

RESUMO

SANTOS, Bruna Bárbara, M.Sc., Universidade Federal de Viçosa, fevereiro de 2007. **Procedimentos híbridos na exploração do espaço de soluções de problemas de natureza combinatória – uma aplicação ao planejamento florestal.** Orientador: Heleno do Nascimento Santos, Co-orientadores: Luiz Aurélio Raggi e Vladimir Oliveira Di Iorio.

O presente trabalho teve como propósito examinar a aplicabilidade de hibridização de técnicas de Programação por Restrições e Metaheurísticas na exploração de espaços de soluções de problemas de natureza combinatória. Para tanto, fez-se uma pesquisa sobre as tendências e aplicações de procedimentos híbridos na literatura. Com base nesta pesquisa, estratégias de hibridização são apresentadas para o problema de planejamento florestal. Um ambiente de otimização capaz de gerar soluções viáveis e pesquisar por melhores soluções foi explorado. Como o gestor florestal defronta-se em seu dia a dia com situações não captadas pelo modelo utilizado, os procedimentos híbridos apresentados devem ser vistos apenas como ferramentas auxiliares no complexo processo de tomada de decisão.

ABSTRACT

SANTOS, Bruna Bárbara, M.Sc., Universidade Federal de Viçosa, February, 2007. **Hybrid procedures in the exploration of the space of solutions of combinatorial problems – an application to the forest planning.** Adviser: Heleno do Nascimento Santos, Co-advisers: Luiz Aurélio Raggi and Vladimir Oliveira Di Iorio.

The present work had as intention to examine the applicability of hybridization of techniques of Constraint Programming and Metaheuristics in the exploration of spaces of solutions of combinatorial problems. For this, was done a research about the trends and applications of hybrid procedures in the literature. Based in this research, strategies of hybridization are presented for the problem of forest planning. An optimization environment capable to generate viable solutions and research for better solutions was explored. As the forest manager faces in his routine situations not caught by the model, the presented hybrid procedures must be seen only as auxiliary tools in the complex process of decision taking.

1. INTRODUÇÃO

Em organizações de diversas áreas, gestores e administradores necessitam de ferramentas que auxiliem o processo de tomada de decisão, uma vez que, normalmente, a quantidade de informações utilizada é muito grande.

O processo de tomada de decisão nas organizações é marcado pela necessidade de busca de soluções ótimas ou sub-ótimas para problemas complexos de natureza combinatória.

Os problemas de planejamento florestal, como muitos de natureza combinatória, são complexos e de difícil formulação.

Segundo Pereira (2004), o gestor florestal gerencia uma grande gama de informações, necessárias para a análise e seleção do melhor plano de manejo possível, que por sua vez é selecionado dentre as alternativas de produção disponíveis.

Para Teixeira (2002), podem-se citar como características inerentes aos problemas de planejamento florestal: o grande porte dos problemas, que são de natureza combinatorial, podendo chegar a milhares de variáveis; a necessidade de satisfazer a vários objetivos simultaneamente, como, por exemplo, uma empresa pode estar interessada em manter os lucros estáveis, aumentar sua fatia do mercado, diversificar produtos, aumentar a produtividade, minimizar as perdas de solo e incrementar os benefícios sociais gerados para a comunidade local; são típicos problemas de programação inteira (0-1), pela conveniência operacional de se manter a identidade das unidades de produção, ou seja, mantê-las sempre com a

mesma área; a natureza estocástica da maioria das variáveis do problema, o que aumenta ainda mais a complexidade.

O planejamento florestal envolve um grande número de variáveis, alto volume de recursos e um número excessivo de restrições, tornando o uso de ferramentas indispensáveis aos gestores florestais no processo de tomada de decisão.

O interesse em buscar metodologias alternativas para abordar o problema de planejamento florestal, que se estendam além dos modelos já conhecidos, é muito grande.

Existem muitas técnicas heurísticas disponíveis para solução de problemas combinatoriais, desde heurísticas menos convencionais até as chamadas metaheurísticas, como *Simulated Annealing* (SA) e Algoritmos Genéticos (AG), já utilizadas, com sucesso, no tratamento desse tipo de problema de otimização.

Neste contexto, a Programação por Restrições (*Constraint Programming*) surge como uma ferramenta alternativa no tratamento deste tipo de problema.

Sucupira (2003) define a Programação por Restrições (PR) como uma tecnologia de programação cuja principal característica é permitir ao programador uma dedicação total à modelagem, tornando oculto o processo de efetiva resolução dos problemas. Como consequência, a programação de restrições tem a capacidade de reduzir o esforço de programação.

De acordo com Baták (1999), a idéia da Programação por Restrições é resolver problemas através da definição das restrições do problema em questão e, conseqüentemente, encontrar uma solução que atenda todas estas restrições.

Recentemente, a PR tem sido aplicada, com sucesso, na busca de soluções para problemas complexos de natureza combinatória, o que motivou o desenvolvimento deste trabalho.

Apesar das aplicações desta técnica em diversos problemas de otimização combinatória, tais como escalonamento (*scheduling*), geração de grade de horário (*time-tabling*) e roteamento, ela ainda tem sido pouco utilizada no meio florestal.

Diante do exposto acima, muitos problemas de otimização florestal são de natureza combinatorial e, portanto, candidatos naturais à solução por esta técnica. Achar o plano de manejo ótimo dentre milhares de intervenções silviculturais, dentro de um horizonte de planejamento longo, em problemas de planejamento florestal é um exemplo (Pereira, 2004).

A principal motivação para este trabalho é buscar possíveis soluções para problemas de planejamento da produção florestal, como o apresentado acima, pois o mesmo ocorre constantemente na rotina de diversas empresas florestais. Contudo, a resolução desse tipo de problema não é trivial. Uma forma de contornar esta complexidade é utilizar uma técnica que já tenha apresentado bons resultados em problemas semelhantes.

Neste trabalho, optou-se pela utilização da técnica de PR, também chamada de *Constraint Satisfaction Problem* (CSP), que viabiliza a especificação e a resolução de problemas semelhantes aos aqui abordados.

1.1. Objetivos

Este trabalho tem como objetivo geral examinar a aplicabilidade do uso combinado das técnicas de Programação por Restrições e/ou de

Metaheurísticas na busca de solução para problemas de natureza combinatória, como uma ferramenta auxiliar no processo de tomada de decisão.

Como objetivos específicos:

- Fazer uma análise das tendências atuais nas aplicações de Programação de Restrições e Metaheurísticas na solução de problemas de natureza combinatória;
- Propor estratégias de combinação dessas metodologias na busca de solução para o problema de planejamento da produção florestal.

1.2. Organização do Trabalho

Esta dissertação está organizada em 5 capítulos, conforme descritos a seguir.

O Capítulo 1 faz uma breve introdução ao trabalho, citando seus objetivos gerais e específicos.

O Capítulo 2 caracteriza o problema de Planejamento da Produção Florestal, a tecnologia de Programação por Restrições e algumas Metaheurísticas, tais como *Simulated Annealing*, Algoritmo Genético e Metaheurísticas Paralelas, onde se faz uma análise do que se conhece sobre o tema, apresentando uma revisão bibliográfica dos referidos temas. Ainda no Capítulo 2, se abordam, de forma resumida, trabalhos que tratam do problema do Planejamento Florestal e trabalhos que utilizam Programação por Restrições e Metaheurísticas com sucesso não somente na área florestal, mas em diversas áreas.

No Capítulo 3, se encontram os processos de hibridização propostos ao Planejamento Florestal, envolvendo a Programação por Restrições, Metaheurísticas e Paralelismo.

No Capítulo 4, são apresentados os resultados encontrados em cada uma das hibridizações.

Enfim, no Capítulo 5, encontram-se as conclusões e recomendações para trabalhos futuros.

2. REVISÃO BIBLIOGRÁFICA

2.1. Planejamento da Produção Florestal

De acordo com Mello (2005), com a implantação da lei dos incentivos fiscais na década de 60, a área de florestas plantadas cresceu enormemente. Os plantios florestais efetuados com esse recurso visavam aumentar a base florestal do País e também captar recursos para o desenvolvimento do setor.

Atualmente, as empresas continuam a implantar reflorestamentos, mas a qualidade final dos produtos e o correspondente valor agregado aos mesmos passaram a ser fatores fundamentais na tomada de decisões das empresas florestais, assim como a preocupação com o meio ambiente, que passou a ser também uma exigência de consumidores externos.

Para isso, existem ferramentas importantes que permitem analisar as informações e a tomada de decisão em função das condições de mercado ou da demanda da própria empresa, como os métodos biométricos que enfocam a prognose do crescimento e da produção da floresta e os métodos de avaliação econômica, de planejamento e de otimização, que consideram critérios econômicos na avaliação de projetos, aliados à programação matemática (Braz, 2001).

Para Rodrigues et al. (2000), a utilização de tecnologias de Sistemas de Informação baseados em computadores, como os Sistemas de Suporte à Decisão (SSD), são indispensáveis ao manejador florestal. Além de propiciar rapidez na formulação e solução de problemas, estas ferramentas propiciam a implementação dos

modelos de métodos científicos para a tomada de decisão no manejo florestal.

Para Silva (2001), atualmente, as pressões de mercados cada vez mais globalizados exigem que as empresas florestais se tornem mais e mais competitivas, devendo produzir produtos de alta qualidade a custos os menores possíveis.

Leite (1994) afirma que o manejo florestal moderno requer integração entre floresta, indústria e mercado, para maximizar o retorno financeiro e, ao mesmo tempo, garantir uma base sustentável do estoque de crescimento da floresta. Embora a exploração de recursos florestais no Brasil em níveis mais organizados e em grande escala seja relativamente recente, não é difícil verificar a grande importância do setor no cenário nacional. Vários setores da economia brasileira estão diretamente ligados ao setor florestal como, por exemplo, o setor siderúrgico a carvão vegetal, o setor moveleiro e o setor de papel e celulose, sendo estes responsáveis por grande parte do PIB nacional e, também, por uma boa fatia dos nossos produtos de exportação.

Os fatores do ambiente, no qual as empresas florestais estão inseridas, influenciam a demanda de produtos com origem florestal e possuem a característica comum de não estarem conectados com a realidade de crescimento biológico das florestas sobre as quais o gestor florestal toma decisões. Em outras palavras, o gerente florestal está pressionado entre duas realidades muito diferentes: de um lado, um comportamento das florestas sujeito a fatores biológicos, climáticos, cujas intervenções têm conseqüências a longo prazo e, de outro lado, uma segunda realidade globalizada, rápida e dinâmica, que exige decisões de curto prazo. Uma floresta não se forma em meses. Mesmo no caso das

florestas de “rápido” crescimento, são necessários pelo menos 5 anos até a colheita, enquanto que a realidade dos fatores citados envolve cenários que podem se reverter em prazos muito menores (Nobre, 1999).

O planejamento criterioso é uma atividade de primordial importância para a obtenção de bons resultados em qualquer tipo de empreendimento econômico (Fatureto, 1997). Neste contexto, a gestão florestal preocupa-se com o processo de tomada de decisões relativas à organização, ao uso e à conservação de recursos florestais. Tais decisões podem ser de longo prazo ou de curto prazo, e resultam em recomendações denominadas planos de manejo florestal (Nobre, 1999).

O grande número de variáveis relacionadas a um processo de planejamento, em especial o florestal, requer grande habilidade e experiência do gerente florestal. Muitas vezes, a excessiva quantidade de variáveis relacionadas a um processo de planejamento pode comprometer o êxito de um empreendimento, se um modelo de planejamento adequado não for utilizado. Na maioria das vezes, o gerente florestal está interessado numa produção sustentável de bens que: atenda as especificações e demandas de mercados; atenda as restrições de capital e operacionais, garantido um emprego regular da mão-de-obra; e apresente um custo mínimo ou um retorno máximo, dentro de um horizonte de planejamento.

Rodriguez. (1996), conceitua gestão florestal como sendo o estudo, desenvolvimento e aplicação de técnicas que ajudam a justificar a escolha de um determinado plano de manejo.

Uma floresta é normalmente dividida em talhões ou unidades de produção. Desta forma, o planejamento florestal implicará na otimização do manejo simultâneo de todas as unidades de produção que constituem a floresta. Estas unidades de produção caracterizam-se pela uniformidade em

termos de produtividade e pela homogeneidade da idade e material genético de que são compostas.

Uma alternativa de manejo, denominada de prescrição, é uma sequência de decisões que ocorrem ao longo de um horizonte de planejamento.

O horizonte de planejamento é um período de tempo estimado durante o qual o empreendimento em análise será operado, podendo ser finito ou infinito. O horizonte é subdividido em períodos, normalmente em anos, e as ações ocorrem no início dos mesmos.

As prescrições representam seqüências preestabelecidas de ações que ocorrerão ao longo do horizonte de planejamento, numa dada unidade de produção (Pereira, 2004).

Segundo Nobre (1999), o problema básico do manejo florestal, é elaborar um plano especificando as diversas intervenções silviculturais que atendem às demandas e restrições definidas para um certo período de tempo. Essas intervenções, incluindo colheitas e plantios, devem necessariamente atender às limitações operacionais e resultar em um fluxo adequado e sustentável de produção.

Uma vez que, para uma dada unidade de produção seja assinalada uma dada prescrição de manejo, ela permanecerá sobre tal prescrição durante todo o horizonte de planejamento.

A escolha da melhor prescrição em cada unidade de produção irá basear-se na melhor alternativa que atenda aos objetivos, às limitações e condições de produção impostas ao projeto.

O problema de planejamento florestal é considerado de difícil solução principalmente devido ao grande número de variáveis envolvidas, e à natureza combinatória do problema quando impostas restrições espaciais e de integridade sobre as unidades de produção.

De acordo com Teixeira (2002), as técnicas de Programação Linear (PL) constituem, tradicionalmente, o instrumental analítico mais utilizado no processo de tomada de decisão no planejamento florestal. Entretanto, apesar da inegável contribuição destas técnicas para a solução de problemas nas mais diversas áreas, deve ser reconhecido que esta metodologia apresenta algumas limitações: trabalha-se apenas com uma função-objetivo e assume-se que as restrições devem ser rigorosamente respeitadas. Esclarece, ainda, que um plano de manejo florestal tem como objetivo especificar as diversas intervenções silviculturais que deverão ocorrer na floresta, para o atendimento de demandas e outras restrições dentro de um certo período de tempo (horizonte de planejamento). Estas intervenções, incluindo colheitas e plantios, devem necessariamente atender às limitações operacionais e resultar em um fluxo adequado e sustentável de produção.

Portanto, a função básica do manejo florestal é conduzir o recurso florestal para atender aos objetivos da empresa.

Um conjunto de unidades de produção constitui o projeto florestal, para o qual se pretende definir uma programação de atividades que atenda ao objetivo global do projeto (Nobre, 1999).

Para cada uma das unidades de produção podem existir centenas de regimes de manejo diferentes. As várias alternativas são obtidas combinando as idades de corte, opções de reforma da floresta, desbastes ou condução de brotação e quaisquer outras características que se queira.

As prescrições podem se referir, por exemplo, às idades de colheita de uma área, seguidas de regeneração. Uma vez que, para uma dada unidade de produção seja assinalada uma dada prescrição de manejo, ela permanecerá sobre tal prescrição durante todo o horizonte de planejamento.

O horizonte de planejamento é subdividido em períodos, e as ações ocorrem no início dos mesmos. É importante salientar que, devido ao grande esforço de cálculo necessário para gerar todas as prescrições, quando se trabalha a nível de uma empresa florestal, faz-se necessária a ajuda de um *software* gerador de prescrições (Teixeira, 2002).

As prescrições (ou regimes de manejo) disponíveis em cada unidade de produção são normalmente avaliadas de acordo com os seus respectivos valores presentes líquidos. São considerados todos os custos e receitas do regime para o horizonte de planejamento (Nobre, 1999; Rodriguez et al., 1996). Contudo, esta avaliação dependerá dos objetivos que serão almejados na resolução do problema de otimização.

A escolha da melhor prescrição em cada unidade de produção irá basear-se na melhor alternativa que atenda aos objetivos, às limitações e condições de produção impostas ao projeto. Portanto, para se atender aos objetivos de produção de todo o sistema florestal, a escolha de prescrições para cada unidade de produção deve ser realizada de forma conjunta, ou seja, escolhe-se para cada unidade uma prescrição tal que, no conjunto, estejam sendo atendidos os objetivos de produção e as limitações operacionais do sistema florestal.

Apesar das inúmeras vantagens da utilização dessas tecnologias para auxiliar o manejador florestal, seu uso tem sido ainda pouco difundido em muitos países, devido a escassez de programas computacionais adequados. Sistemas de suporte à decisão para

solução de problemas de planejamento florestal em larga escala como o FORPLAN, Timber RAM e MAX MILLION, são alguns exemplos dos primeiros sistemas desenvolvidos com sucesso. Apesar do sucesso em território americano, a utilização desses sistemas não se consolidou em nosso país. Escassez de especialistas para entender e manipular estes sistemas, interface pouco amigável e a falta de adequação para a realidade brasileira foram algumas barreiras que desencorajaram a sua utilização no Brasil (Rodrigues et al., 2000).

Novos sistemas de suporte à decisão têm sido propostos na tentativa de solucionar as limitações dos sistemas precedentes. Por exemplo, Rodrigues et al. (2000) apresentou o protótipo de um sistema de suporte à decisão, denominado SYSFLOR (Sistema de Suporte à Decisão Otimizada em Planejamento Florestal), que incorpora novos paradigmas na solução de problemas de planejamento florestal. A proposta central do sistema é permitir maior flexibilidade na formulação e na solução de diversos problemas de planejamento florestal que, até então, não era tratada pela maioria dos sistemas vigentes.

Para Rodrigues et al. (2000), apesar do grande avanço da PO na área florestal, com a evolução dos modelos de Programação Linear (PL) para modelos de Programação Inteira (PI) e Programação com Múltiplos Objetivos (PMO), e com relatos de uma infinidade de trabalhos enfocando a utilização desses modelos no suporte à tomada de decisão no planejamento florestal, a solução de problemas de grande porte tem sido ainda bastante limitada, principalmente no Brasil. A complexidade dos problemas, o volume de dados envolvidos e a carência de tecnologias adequadas são alguns fatores que dificultam a utilização desses modelos em larga escala.

Apesar dos grandes avanços na pesquisa e a utilização em grande escala de variados modelos em diversos setores da sociedade, a utilização dessas ferramentas no manejo de recursos florestais é ainda bastante incipiente, principalmente no Brasil. O desconhecimento por parte de alguns gerentes dos potenciais dessas ferramentas para auxiliar o manejador florestal é ainda um entrave na sua adoção. Entretanto, a carência de interfaces computadorizadas para viabilizar as aplicações desses modelos em grande escala é, certamente, o fator que atualmente mais dificulta as aplicações em problemas de grande porte. A utilização de sistemas computadorizados é indispensável para viabilizar o processamento e armazenamento de volume excessivo de dados, gerar novas informações, manipular de forma suave os variados modelos utilizados em diferentes situações, pesquisar as soluções do problema e oferecer recursos para apresentação de resultados (Rodrigues et al., 2000).

Os principais trabalhos envolvendo a utilização de algoritmos exatos, como o algoritmo *branch-and-bound* para solução de problemas de natureza combinatória, mostraram que estes são eficientes apenas em problemas de pequeno porte (Pereira, 2004).

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 (Rodrigues, 2001).

Segundo Rodrigues et al. (2003), as principais metaheurísticas com grande potencial de aplicação na solução de problemas de planejamento florestal são: Algoritmos Genéticos (AG) (Holland, 1975; Goldberg, 1989), Busca Tabu (BT) (Glover, 1977) e *Simulated Annealing* (SA) (Kirkpatrick et al., 1983; Cerny, 1985).

De acordo com Pereira (2004), 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 & Moccellini, 2000).

Teixeira (2002) desenvolveu um trabalho utilizando algoritmos evolucionários em otimização sob múltiplos objetivos, como uma ferramenta auxiliar no processo de tomada de decisão associado ao planejamento da produção florestal, implementando o algoritmo evolucionário SPEA (*Strength Pareto Evolutionary Algorithm*) para a solução heurística do modelo proposto.

O trabalho de Rodrigues et al. (2003) teve como objetivo desenvolver e testar um algoritmo com base na metaheurística Busca Tabu (BT), para a solução de problemas de gerenciamento florestal com restrições de inteireza.

Buscando novas alternativas na área, Pereira (2004) aplicou *Simulated Annealing* na solução de problemas de planejamento florestal multiobjetivo, propondo um modelo de otimização sob critérios múltiplos e implementando um algoritmo para a solução do modelo, o PSA (*Pareto Simulated Annealing*), uma variação do *Simulated Annealing* uniobjetivo, para a solução heurística do modelo sob a visão multiobjetivo.

Vista a necessidade do uso de uma ferramenta para auxílio no planejamento florestal e a complexidade do problema, o interesse por inovações na área é constante, motivando os pesquisadores a buscarem

novas tecnologias e ferramentas que possam ser úteis na solução do problema.

2.2. Programação por Restrições

As primeiras idéias relacionadas à Programação por Restrições (PR) surgiram na área de Inteligência Artificial (IA) nos anos sessenta e setenta (Sutherland, 1963; Fikes 1968; Montanari, 1974; Waltz, 1975)

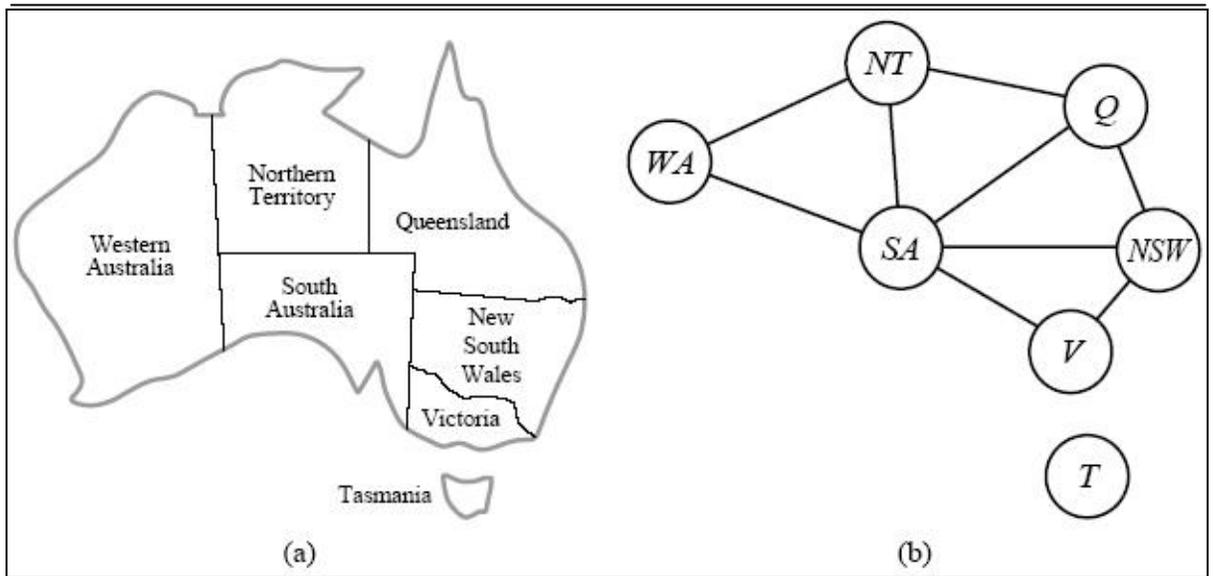
Segundo Tavares (2000), os avanços tecnológicos da época (décadas de 60 e 70) focalizavam-se na representação e manipulação explícita de restrições em sistemas computacionais. Apenas nas décadas de 80 e 90 desenvolveu-se uma crescente conscientização de que estas idéias podiam fornecer a base para uma abordagem poderosa à temática da programação, modelagem e resolução de problemas (Steele, 1980). Conseqüentemente foram desenvolvidos esforços para explorar estas idéias e unificá-las numa única estrutura conceitual (Jaffar & Lassez, 1987).

As qualidades da programação de restrições citadas por Sucupira (2003), apoiadas em uma forte fundamentação teórica e aliadas à eficiência dos sistemas existentes para a prática da programação de restrições, têm resultado num grande sucesso, tornando essa tecnologia escolha freqüente para o tratamento de diversas classes de problemas, como por exemplo, o problema do caixeiro viajante, das N-Rainhas, da coloração de mapas, entre outros.

De acordo com Baták (1999), nos últimos anos, a Programação por Restrições atraiu a atenção de pesquisadores de diversas áreas, devido ao seu alto potencial para resolução de problemas complexos do mundo real.

Um algoritmo é completo se existir solução para ele. Possui profundidade se cada resultado retornado por ele for uma solução para o problema proposto em que todos os valores atribuídos às variáveis satisfaçam todas as restrições do problema. Completude e profundidade são características desejáveis dos algoritmos de CSP. A maioria dos algoritmos encontrados em Tsang (1993) alcança essas características. Como nem todos os problemas são solúveis, os algoritmos incompletos, mas eficientes, às vezes são considerados aceitáveis, como o algoritmo de subida na montanha (TSANG, 1993, p. 31).

Russel e Norving (2003, p. 138) apresentam como exemplo um mapa da Austrália que precisa ser pintado, mostrado na 1 (a). Para facilitar a definição do problema, a Figura 1 (b) possui um grafo com as fronteiras das regiões. Para colorir o mapa, podem-se utilizar as cores vermelha, verde e azul, e nenhuma região vizinha pode ter a mesma cor. Como variáveis, definem-se as regiões: WA, NT, SA, Q, NSW, V e T. O domínio de cada variável são as cores {vermelha, verde, azul}. A restrição é que cada região vizinha não pode possuir a mesma cor. Por exemplo, entre as várias combinações possíveis para WA e NT estão {(vermelha; verde); (vermelha; azul); (verde; vermelha); (verde; azul); (azul; vermelha); (azul; verde)}. As restrições também podem ser representadas por $WA \neq NT$, desde que a expressão seja reconhecida pelo algoritmo a ser desenvolvido. Para esse problema, existem várias soluções possíveis, como: {WA = vermelha; NT = verde; Q = vermelha; NSW = verde; V = vermelha; SA = azul; T = vermelha}.

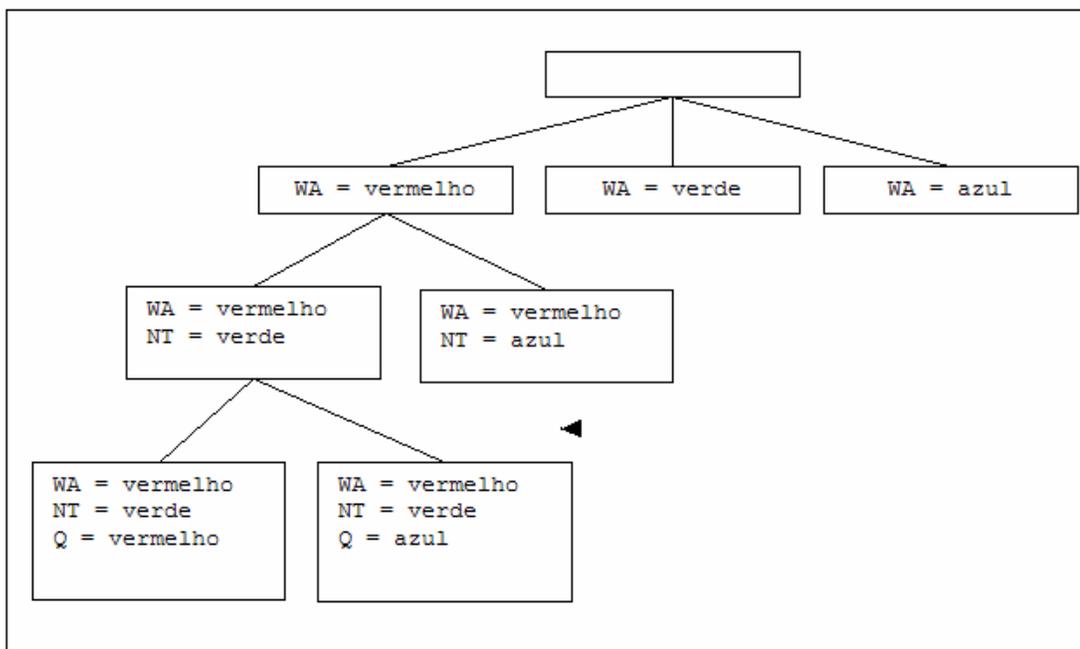


Fonte: Russell e Norving (2003, p. 138).

Figura 2 – Mapa da Austrália

Há diversos algoritmos que podem resolver problemas de CSP, como o algoritmo por *backtracking*, o de checagem regressiva (*forward checking*), o de mínimos conflitos (*Min-Conflicts*) e os híbridos, que são constituídos da combinação de diferentes técnicas (TSANG, 1993, p. 35).

Dos algoritmos existentes, como exemplo, descreve-se o de busca por *backtracking*. Busca por *backtracking* é o termo usado para as buscas em profundidade de primeira escolha (*depth-first search*). Em uma árvore de busca, como a mostrada na Figura 2, que baseia-se no mapa da Figura 1, o algoritmo gera todos os estados sucessores do nodo atual e aprofunda-se no nodo da esquerda, gerando seus sucessores e aprofundando-se. Ao encontrar um valor não-consistente, volta ao nível superior e aprofunda no nodo da direita daquele de valor não-completo. Caso não existam nodos à direita, volta mais um nível acima e faz a mesma verificação sucessivamente (RUSSEL & NORVING, 2003, p. 142).



Fonte: Russel e Norving (2003, p. 142).

Figura 2 – Parte de uma árvore de busca

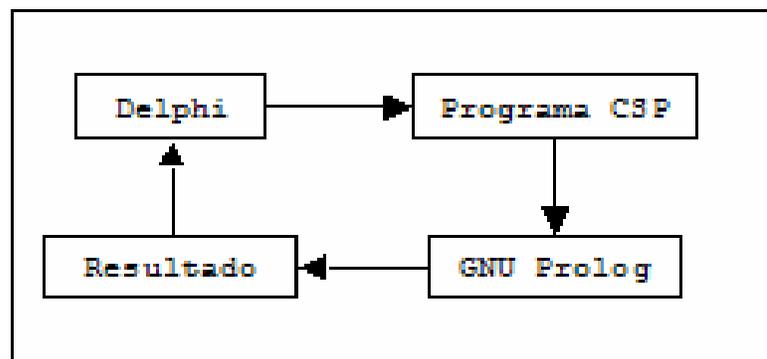
Confirmando esta tendência, a PR tem sido utilizada em sistemas para logística de distribuição.

O objetivo destes sistemas é estabelecer rotas para entregas e coletas de mercadorias por vias rodoviárias. As rotas sempre buscam o melhor caminho para que os veículos tenham melhor aproveitamento e para que suas capacidades não sejam excedidas.

No sistema desenvolvido por Rosa (2005), foi utilizada a combinação de conceitos logísticos, técnicas e linguagens de programação e ferramentas de desenvolvimento. A linguagem de programação *Prolog* combinada com o interpretador *GNU Prolog* resolve o CSP do problema proposto. A linguagem de programação *Object Pascal* e o ambiente de desenvolvimento *Delphi*

constituem o sistema, a geração de códigos fontes na linguagem de programação *Prolog* e a integração com a ferramenta *GNU Prolog*.

A figura abaixo (Figura 1) apresenta o diagrama com a integração entre o *Delphi* e o *GNU Prolog*, de acordo com o trabalho desenvolvido por Rosa (2005).



Fonte: Rosa (2005)

Figura 3 – Integração entre o *Delphi* e o *GNU Prolog*

O trabalho de Ciré et al. (2005), utiliza a técnica de Programação por Restrições em Domínios Finitos proposta por Marriot & Stuckey (1998) na construção do conjunto de jornadas viáveis para solução Problema de Programação de Viagens (PPV). O objetivo é elaborar o agendamento dos veículos, as escalas da tripulação, e definir os horários de partida de uma determinada linha de ônibus, atendendo todas as restrições operacionais, minimizando os tempos de espera entre as viagens e otimizando a utilização dos recursos mais críticos, como quantidade de ônibus e horas extras de funcionários.

Já em Sucupira (2003), a PR foi utilizada na Distribuição de horários para o Departamento de Ciência da Computação do Instituto de Matemática

e Estatística da Universidade de São Paulo. O objetivo é a construção de grades horárias de aulas, atendendo as restrições naturais, além de restrições impostas pelo usuário, buscando a otimização da escala de horários. A modelagem do problema proposto foi feita utilizando o ambiente *Mozart-OZ* (Muller, 2001).

Outro exemplo da utilização de PR é o trabalho de Marchioro (2003), que aborda o problema de escalonamento de tarefas com restrição de recursos de mão-de-obra e ferramental aplicado à área de planejamento de manutenção industrial. A estratégia de solução proposta pela autora está baseada na aplicação de programação lógica por restrições para domínios finitos (*constraint logic programming for finite domain - CLPFD*) implementada no software de programação lógica por restrições SICStus Prolog combinando assim a natureza declarativa da programação em lógica com a eficiência dos métodos de resolução de problemas com satisfações de restrições que se mostraram bastante eficazes na resolução de problemas de natureza combinatória.

Baták (1999) afirma que restrições existem nas mais diversas áreas, formalizando as dependências entre o mundo físico e suas abstrações matemáticas, de forma natural e transparente.

Uma restrição é definida como sendo uma relação entre os valores que as variáveis podem assumir (Choose Technologies, 2003).

As restrições matemáticas especificam de uma forma clara e precisa as relações entre diversos parâmetros desconhecidos (variáveis), cada um tomando um dado valor no domínio considerado. As restrições restringem os valores possíveis que as variáveis podem tomar e representam alguma informação (parcial) sobre as variáveis em jogo.

Ciré et al. (2005) afirmam que a programação por restrições é uma ferramenta poderosa para a modelagem de problemas combinatórios, principalmente aqueles relativos a planejamento e agendamento. Com ela, é possível criar um modelo definido por variáveis e relações restritivas entre elas, e requisitar ao ambiente resolvidor valores que satisfaçam as restrições desejadas.

Um problema de restrição é constituído de um conjunto de variáveis e um conjunto de restrições nos valores que estas variáveis podem assumir. O conjunto de valores possíveis para cada variável é chamado de domínio da variável (Choose Technologies, 2003).

Ao se modelar um problema nesse ramo da programação por restrições, são necessários três passos: (i) definir as variáveis e o domínio de cada uma; (ii) definir as restrições, utilizando-se, entre outras, operações aritméticas e booleanas; e (iii) realizar a busca pela solução iterativamente, atribuindo valores às variáveis respeitando as restrições e domínios definidos (Ciré et al., 2005).

A propagação de restrições, essencialmente, é utilizar as restrições de forma construtiva para deduzirmos novos valores e detectarmos impossibilidades o mais rápido possível, através da redução de domínios.

A programação por restrições tem obtido grande sucesso em problemas complexos de otimização combinatória. Neste caso, além da simplificação da modelagem e do bom desempenho devido à propagação de restrições, a programação de restrições tem a oferecer uma grande facilidade no controle das técnicas de busca.

A programação por restrições é formalmente definida como um conjunto de variáveis, X_1, X_2, \dots, X_n , e um conjunto de restrições

(*constraints*), C_1, C_2, \dots, C_n . Cada variável X_i tem domínio D_i , não-vazio e de valores possíveis. Cada restrição C_i envolve algum subconjunto de variáveis e especifica as combinações dos valores possíveis para esse subconjunto. Um estado do problema é definido por atribuição dos valores a algumas ou a todas as variáveis $\{X_1 = V_1, X_2 = V_2, \dots, X_n = V_n\}$. Uma atribuição que não viole nenhuma das restrições é chamada de atribuição consistente ou legal. Uma atribuição completa possui todas as variáveis mencionadas, e uma solução é uma atribuição completa que satisfaz a todas as restrições (Russel & Norving, 2003).

Rosa (2005) citando Tsang (1993), afirma que o conjunto de variáveis, o conjunto de restrições e o domínio D_i devem ser conjuntos finitos, pois os algoritmos eficientes exploram essas características. Um algoritmo é completo se existir solução para ele. Possui profundidade se cada resultado retornado por ele for uma solução para o problema proposto em que todos os valores atribuídos às variáveis satisfaçam todas as restrições do problema. Completude e profundidade são características desejáveis dos algoritmos de CSP. A maioria dos algoritmos encontrados em Tsang (1993) alcança essas características. Como nem todos os problemas são solúveis, os algoritmos incompletos, mas eficientes, às vezes são considerados aceitáveis, como o algoritmo de subida na montanha.

2.3. Metaheurísticas

Blum et al. (2005), numa introdução às técnicas metaheurísticas, definem Metaheurística como uma técnica de busca de soluções aproximadas, não necessariamente ótimas, capaz de obter solução razoável para problemas complexos de natureza combinatória em tempo relativamente curto. Ao contrário da heurística, esta técnica se aplica a

diversas classes de problemas de natureza combinatória, sendo que o termo Meta diz respeito a uma abordagem de alto nível. Segundo esses autores:

- ◆ Metaheurísticas são estratégias que guiam o processo de busca.
- ◆ O objetivo é explorar de forma eficiente o espaço de busca, a fim de encontrar soluções quase ótimas (ou sub-ótimas).
- ◆ Algoritmos metaheurísticos são aproximativos e usualmente não determinísticos.
- ◆ Devem incorporar mecanismos de fuga de busca local.
- ◆ Seus conceitos básicos são descritos em alto nível.
- ◆ Não são específicos a um dado problema.
- ◆ Devem usar conhecimento específico de um dado problema na forma de heurística para auxiliar a exploração do espaço de busca.

Classificam, ainda, as metaheurísticas como:

a) Métodos de Trajetória (*Simulated Annealing*, Busca Tabu, *GRASP*, etc.);

b) Métodos Baseados em População (Algoritmo Genético, Otimização Colônia de Formiga);

c) Metaheurísticas Paralelas;

d) Metaheurísticas Híbridas.

Em seguida, descrevem-se exemplos de técnicas associadas a cada categoria.

2.3.1. Método de Trajetória: *Simulated Annealing*

Algoritmo *Simulated Annealing*:

$s \leftarrow$ solução inicial gerada segundo algum critério

$K \leftarrow 0$

$T_k \leftarrow$ Temperatura Inicial

Enquanto critério de parada não satisfeito **Faça**

$s' \leftarrow$ solução vizinha gerada aleatoriamente em $N(S)$

Se $(f(s') < f(s))$ **então**

$s \leftarrow s' \{ s' \text{ substitui } s \}$

Senão

Aceita-se s' como nova solução, com probabilidade p sendo função de T_k e s

FimSe

$T_k \leftarrow \alpha * T_k \quad \{ \text{redução de temperatura} \}$

$k \leftarrow k+1$

Fim Enquanto

Saída *melhor solução encontrada*

Como é característico do funcionamento das Metaheurísticas, observa-se uma fase de busca local (intensificação) seguida de uma fase de escape da busca local (diversificação), sendo esta última fase identificada, nesta técnica, pela aceitação de uma solução pior com alta probabilidade no início da busca, probabilidade que diminui à medida que a busca avança. A vizinhança de uma solução s é definida por meio de um movimento feito a partir de s .

2.3.2. Método Baseado em População: Algoritmo Genético

Algoritmo Genético:

$P \leftarrow$ População inicial gerada

Avaliar(P)

Enquanto *critério de parada não satisfeito* **Faça**

$P' \leftarrow$ *Recombinar* (P)

$P'' \leftarrow$ *Mutar* (P')

Avaliar (P'')

$P \leftarrow$ *Selecione* (P, P'')

Fim Enquanto

Saída *melhor solução encontrada*

Também aqui, observa-se uma fase de busca local seguida de uma fase de escape da busca local, sendo esta última fase identificada pelo mecanismo de mutação, após a fase de recombinação.

2.3.3. Metaheurísticas Paralelas

Consistem em procedimentos capazes de explorar o espaço de soluções de problemas de natureza combinatória por meio buscas simultâneas em subespaços de soluções ou por meio de computação paralela que envolve questões de hardware e de software. Alba et al. (2005) descrevem relacionamentos entre Metaheurísticas e Paralelismo, com a exposição de aplicações em diversos tipos de problemas.

2.3.4. Metaheurísticas Híbridas

Blum et al. (2005), no capítulo sobre Introdução a Técnicas Metaheurísticas (Alba, 2005), a hibridização de metaheurísticas pode ocorrer por meio de:

- a) mudança de componentes entre metaheurísticas;
- b) pesquisa cooperativa.

Ainda em Alba (2005), Cotta et al. discorrem sobre a hibridização de metaheurísticas, apresentando notas históricas associadas a esses procedimentos, uma classificação das metaheurísticas híbridas e uma extensão para o estudo de metaheurísticas híbridas paralelas e de suas implementações. Também, nesta obra, Nebro et al. escrevem um capítulo sobre Otimização Multiobjetivo Paralela, atingindo o ápice de todo um elenco de técnicas capazes de explorar complexos espaços de soluções que encontram-se associados a problemas de natureza combinatória.

CIRÉ et al. (2005) usam a hibridização de Programação de Restrições com Metaheurísticas Paralelas para a solução de um problema de escalonamento integrado de veículos e tripulação. Nesta aplicação, soluções viáveis são geradas com o uso de Programação por Restrições e o espaço de soluções é explorado por meio de Metaheurísticas Paralelas.

Ochi et al (2006) usam um algoritmo evolutivo híbrido para a formação de células de manufatura em sistemas de produção, alterando a estrutura básica de funcionamento do Algoritmo Genético convencional. Propõem as seguintes alterações: a população inicial não é gerada de forma aleatória, mas por meio de um procedimento heurístico que garanta melhores indivíduos; um novo mecanismo de seleção e cruzamento foi usado de modo a cruzar melhores indivíduos e em número maior ou igual a

dois; um método eficiente de busca local é proposto para ser usado sobre parte da população a cada iteração, realizando uma busca intensiva na vizinhança das melhores soluções encontradas pelo algoritmo evolucionário, com o objetivo de obter um número maior de ótimos locais.

Inúmeras são as aplicações desses procedimentos na busca de solução para problemas de natureza combinatória, como pode ser observado, por exemplo, em Alba et al. (2005).

3. MATERIAIS E MÉTODOS

O modelo de Planejamento Florestal pode ser colocado, conforme descrito em Pereira (2004), dentre outros objetivos, como sendo o de minimizar a função objetivo representada pelo Valor Presente Líquido (VPL) global da exploração florestal:

$$\text{Max VPL} = \sum_{i=1}^M \sum_{j=1}^N L_{ij} X_{ij}$$

em que:

VPL = valor presente líquido (VPL) global;

L_{ij} = valor presente líquido (VPL) da unidade de produção i , manejada sob a prescrição j ;

X_{ij} = variável binária, que ao assumir o valor 1 significa cortar a unidade de produção i sob a prescrição j , e 0 significa o contrário;

M = número total de unidades de produção; e

N = número total de prescrições da i -ésima unidade de produção.

As restrições associadas são assim definidas:

- **Restrição de manejo**

Esta é a restrição que determina que cada unidade florestal seja manejada por apenas uma prescrição, sendo definida por:

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

em que:

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

0 = não-assinalamento da alternativa j na unidade de manejo i.

1 = caso contrário.

- **Imposição de cotas anuais de produção**

Esta é uma das restrições de maior importância, uma vez que a principal meta do empreendedor é a produção periódica de volume de madeira de modo a atender a uma demanda preestabelecida por uma fábrica ou atender a uma demanda contratual do mercado vigente.

Considerando-se V_K o volume total removido no período K, tem-se:

$$V_K \leq \sum_{i=1}^M \sum_{j=1}^N V_{ijk} X_{ij} \quad \{K = 0, 1, \dots, H - 2\}$$

em que:

V_{ijk} = volume produzido pela i-ésima unidade de produção utilizando a j-ésima alternativa de manejo no início do período k;

V_K = volume total (m^3) removido no período k, representando a demanda a ser atendida no período k; e

H = horizonte de planejamento.

- **Restrição de capacidade de reforma**

Essa restrição visa estabelecer uma capacidade máxima de reforma que a empresa florestal é capaz de realizar a cada período. Considerando-se CRK a capacidade de reforma preestabelecida para o período k, então:

$$CR_k \geq \sum_{i=1}^M \sum_{j=1}^N A_{ijk} X_{ij} \quad \{k = 0, 1, \dots, H-2\},$$

em que:

A_{ijk} = área da *i*-ésima unidade de produção, manejada sob a *j*-ésima prescrição, no início do período k e que esteja sob o regime de manejo de reforma.

Na exploração do espaço de soluções inerentes ao problema de planejamento florestal, representado pelo modelo acima, algumas estratégias de busca são expostas a seguir, com base em procedimentos híbridos.

3.1. Hibridização usando Programação por Restrições e Metaheurística.

A técnica de Programação por Restrições é usada com vistas a gerar soluções viáveis para o problema de planejamento florestal. A partir das soluções viáveis geradas, usa-se uma metaheurística para gerar novas soluções que preservem a viabilidade e que sejam melhores.

Atualmente, existem algumas ferramentas e linguagens que solucionam um CSP. A seguir, apresenta-se a linguagem *Easy abstract Constraint Optimization Programming Language* (EaCL) as ferramentas *ZDC*

e *ZDC Direct*.

A combinação da linguagem EaCL com a ferramenta ZDC foi utilizada no processo de geração das prescrições do problema de planejamento florestal.

De acordo com Rosa (2005), citando Bradwell et al (2004), a comunidade de CSP, composta por pesquisadores da Universidade de Essex do Reino Unido, a partir do projeto *Computer-Aided Constraint Programming* (CACP), desenvolveu uma ferramenta que soluciona todo o processo de programação por restrições. A ferramenta suporta tarefas da formulação e da entrada do problema. Os problemas são modelados na linguagem declarativa EaCL. A comunidade teve o cuidado de desenvolver uma interface amigável cuja entrada das informações é bastante simples.

Com a ferramenta *ZDC*, o fluxo principal do desenvolvimento começa com a entrada da definição do problema na linguagem EaCL.. Os dados necessários para a solução do problema podem ser importados de um arquivo do software *Excel*. A descrição do problema pode ser feita a partir da ferramenta *ZDC*, na sua *interface* gráfica, ou na ferramenta *ZDC Direct*, a partir de um arquivo texto. (Bradwell et al, 2004)

Podemos definir a EaCL como uma linguagem declarativa de alto nível para descrever CSP's e CSOP's.

A definição de um problema em EaCL é dividida em 4 seções, conforme apresentado a seguir na Figura 4:

```

Problem:NomeDoProblema
{
    Data
    {
        //Dados que definem uma instância do problema
    }

    Domains
    {
        //Declaração dos domínios
    }

    Variables
    {
        //Declaração das variáveis e arrays de
        //variáveis, com um dado domínio
    }

    Constraints
    {
        //Declaração das restrições
    }

    Optimisation
    {
        //Minimizar o maximizar alguma função
    }
}

```

Figura 4 – O esqueleto de um problema em EaCL.

Na Figura 4, apresentam-se a especificação do CSP, utilizando a linguagem EaCL, e a ferramenta *ZDC Direct*, para o problema proposto na Figura 1. As cidades e as cores foram representadas como números. Os valores correspondentes aos conjuntos são: cidades {0 = WA; 1 = NT; 2 = Q; 3 = NSW; 4 = V; 5 = SA; 6 = T} e cores {0 = vermelho; 1 = verde; 2 = azul}.

```

Problem:cordomapa
{
  Data
  {
    NumCidade := 7;
  }
  Domains
  {
    IntDom Cores=[0,2];
  }
  Variables
  {
    IntVar Cidades[7]::Cores;
  }
  Constraints
  {
    Forall ( i in [0 .. 4] )
    {
      Cidades[i] <> Cidades[i+1];
      Cidades[i] <> Cidades[5];
    }
  }
}

```

Figura 5 - Especificação do CSP para o problema da pintura do mapa

A solução obtida, ao se executar o programa na ferramenta *ZDC Direct*, foi {WA = vermelho; NT = verde; Q = vermelho; NSW = verde; V = vermelho; SA = azul; T = azul}

O termo usado para definir um CSP que utiliza otimização em sua solução é *Constraint Satisfaction Optimization Problems* (CSOP) que, por sua vez, possui a função de encontrar soluções ótimas para os problemas. Um CSOP é definido por um CSP juntamente com uma função de otimização f que mapeia todas as soluções para valores numéricos. A tarefa do CSOP é procurar a solução ótima (custo máximo ou custo mínimo) para o valor de f

conforme a função f utilizada. (Tsang, 1993).

Bradwell et al (2004) resolvem um CSOP para o problema do caixeiro-viajante. O problema é definido por um viajante que deve visitar algumas cidades e retornar para a cidade de origem. Cada cidade pode ser visitada somente uma vez, e todas as cidades devem ser visitadas. O objetivo é minimizar a distância a ser percorrida. A formulação da solução utilizando a linguagem de programação EaCL para o problema do caixeiro-viajante é dada pela Figura 6.

```

1  Problem:TSP
2  {
3    Data
4    {
5      distances := [0,60,57,120,200,
6                    60,0,115,175,210,
7                    57,115,0,0,100,260,
8                    120,175,100,0,120,
9                    200,210,260,120,0] ;
10   Max := 5 ;
11   }
12   Domains
13   {
14     IntDom Num = (0,Max-1) ;
15   }
16   Variables
17   {
18     IntVar Cities[Max]::Num ;
19   }
20   Constraints
21   {
22     AllDifferent(Cities) ;
23   }
24   Optimisation
25   {
26     Minimise
27     (
28       Sum(
29         [ distances[(cities(x)*Max)+cities[x+1]]
30         | x in [0 .. Max-2 ]
31         ++ [ distances[(cities[Max-1]*Max)+cities[0]] ] )
32     ) ;
33   }

```

Fonte: Bradwell et al (2004).

Figura 6 – Problema do caixeiro-viajante descrito na linguagem de programação EaCL

Na Figura 6, no intervalo entre as linhas 3 (três) e 11 (onze), são declarados os dados utilizados no problema. A constante *Max* define o

número de cidades a serem visitadas no problema. A lista *distances* define as distâncias geométricas entre cada uma das cidades.

Ainda na Figura 6, a seção *Domain* define um único domínio inteiro *Num*, que pode receber os valores entre 0 (zero) e *Max-1* (um), ou seja, entre zero e quatro. A solução para o problema é apresentada pela variável *Cities*, que é representada por um vetor de restrições. Cada posição de *Cities* contém um número da cidade e o valor atribuído para ela. Para garantir que nenhuma cidade seja visitada mais de uma vez e que todas as cidades sejam visitadas, é aplicada ao vetor *Cities* a restrição *AllDifferent*. A partir da linha 24 (vinte e quatro), é especificada a otimização para o problema.

A otimização para o problema apresentado na Figura 6 tem por objetivo minimizar a distância a ser percorrida pelo caixeiro-viajante em seu percurso. A expressão definida na linha 29 (vinte e nove) gera uma lista das distâncias entre cada cidade na seqüência em que ela é armazenada pelo vetor *Cities*. As distâncias são os valores que foram atribuídos para a lista *distances*. Para a geração da lista, são utilizados os valores do vetor *Cities* como índices. Essa lista é concatenada, utilizando o operador ++, com a expressão disposta na linha 30 (trinta), onde é definida a cidade final do trajeto. A soma de todos os valores armazenados na lista resulta no custo final da viagem (BRADWELL et al, 2004).

A ferramenta ZDC oferece as seguintes opções de solvers: GFC (*General Forward Checking*), GLS (*Guided Local Search*), GA (*Genetic Algorithm*), *Tabu Search* (Busca Tabu), *Expert Solver* e *External Solver*.

Para a modelagem do gerador de prescrições, os seguintes passos foram seguidos:

- 1º passo: Especificação do problema;

- 2º passo: Modelagem das restrições (envolve a definição das variáveis, domínios e restrições do problema, baseados na especificação do problema);
- 3º passo: Declaração do problema na linguagem EaCL;
- 4º passo: Solucionar o problema.

A Figura 7 apresenta uma visão da ferramenta ZDC, em sua interface gráfica.

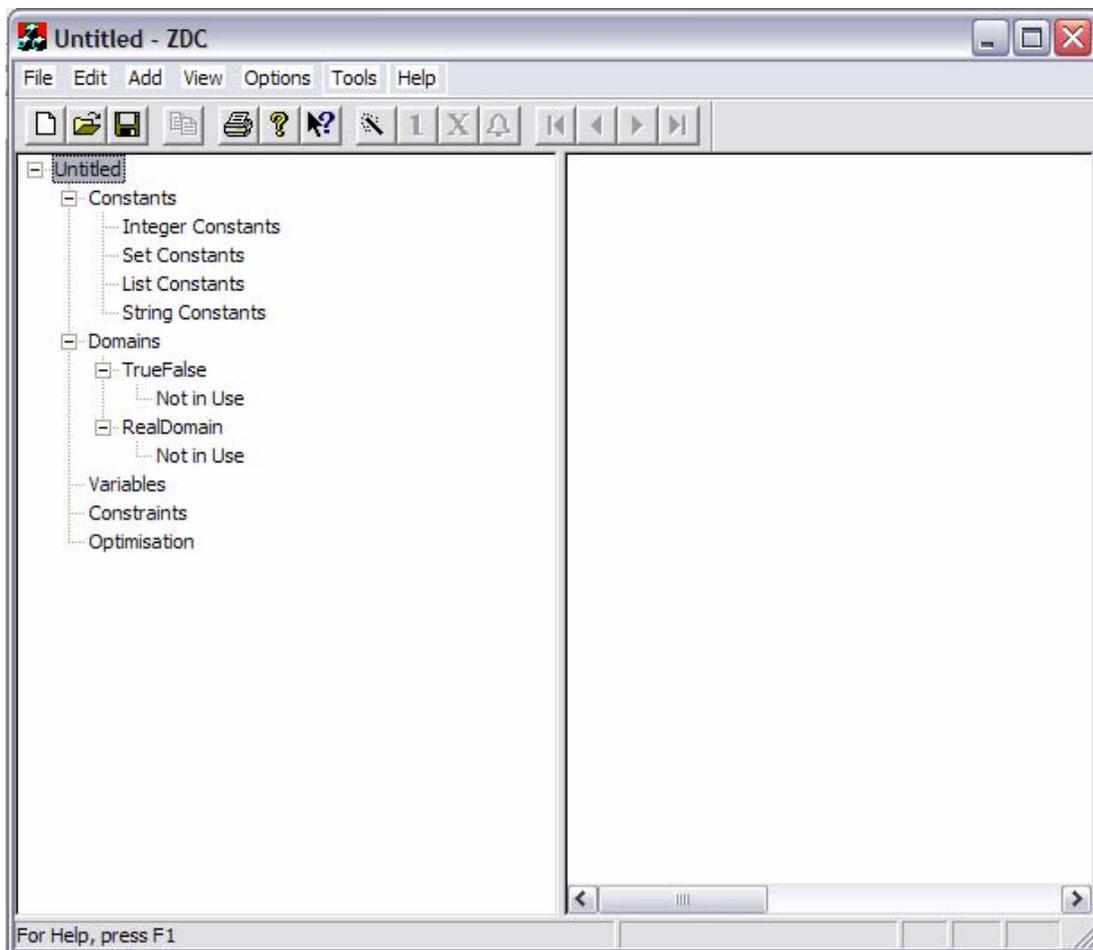


Figura 7 – Visão da ferramenta ZDC, em sua interface gráfica

3.2. Hibridização da Metaheurística Algoritmo Genético.

Com base no trabalho de Ochi et al. (2006), a estrutura convencional do Algoritmo Genético será modificada, de modo a permitir um processo evolutivo melhor.

A hibridização da Metaheurística Algoritmo Genético não foi desenvolvida totalmente, ficando aqui apenas como sugestão para trabalhos futuros.

4. RESULTADOS E DISCUSSÃO

4.1. Hibridização usando Programação por Restrições e Metaheurística.

O uso da Programação por Restrições na geração das prescrições do para o Problema de Planejamento Florestal utilizando a ferramenta ZDC e a linguagem EaCL, como proposto, resultou numa forma alternativa de geração para as prescrições das unidades de produção, diferente das formas até então utilizadas.

A Figura 8, apresentada a seguir, mostra a ferramenta ZDCDirect, gerando as prescrições do problema teste proposto, considerando como idade mínima de corte 2 anos, idade máxima de corte 4 e horizonte de planejamento de 12 anos.

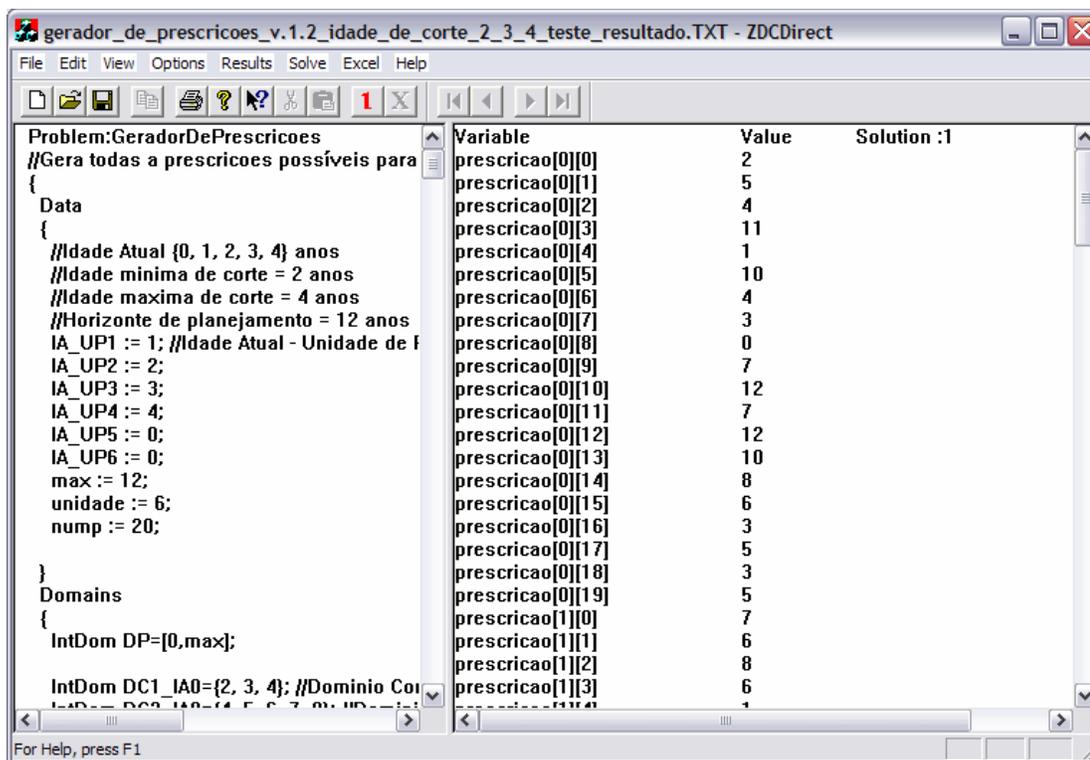


Figura 8 – Tela da ferramenta ZDC gerando as prescrições do problema teste proposto

Apesar de não utilizar as funcionalidades de otimização disponíveis na própria ferramenta ZDC, tornou-se clara a facilidade de modelagem e descrição de um CSP através da linguagem EaCL.

O primeiro passo na resolução de qualquer problema consiste na determinação da abrangência ou domínio do problema. O conjunto de todos os possíveis estados do problema, onde se encontra a solução ainda desconhecida, denomina-se espaço de busca. Utilizando-se a técnica de programação por restrições, são formuladas condições que diminuem substancialmente a quantidade de buscas necessárias para se encontrar a solução do problema.

Considerando-se a complexidade e porte dos problemas de Planejamento Florestal, a redução do tempo gasto para se encontrar a solução de um problema pode melhorar consideravelmente o desempenho das aplicações.

Baseando-se em trabalhos desenvolvidos nas áreas de escalonamento e roteamento, é possível combinar o uso da ferramenta ZDC e outros ambientes de programação visual, como *Delphi* e *Visual Basic*.

De acordo com Rosa (2005), comparando-se a técnica de programação por restrições com outros procedimentos tradicionais para solução de problemas de otimização discreta, como a programação linear, notamos que a primeira possui algumas vantagens qualitativas.

A primeira delas diz respeito à representação do problema, que no caso da programação por restrições é muito mais compacta - exigindo um número muito menor de variáveis - com melhorias no desempenho da aplicação.

O segundo ponto relaciona-se ao processo de propagação das restrições, que permite também um desempenho melhor ao eliminar mais rapidamente combinações que não solucionam o problema, e, portanto agilizando a busca pela solução.

E por último, o que talvez seja a principal vantagem, a programação por restrições permite uma representação mais direta de alguns problemas do que, por exemplo, a programação linear. Essa última, por estar atrelada aos rígidos algoritmos implementados, necessita de uma estrutura que, muitas vezes, impõe uma adaptação das variáveis para a representação do problema.

A programação por restrições, por ser implementada através uma linguagem de programação, permite a elaboração de algoritmos especialmente dedicados a cada tipo de questão, conseguindo priorizar os pontos de maior importância, identificar pontos de infactibilidade e propor caminhos alternativos segundo o entendimento e intervenção dos especialistas. Nesse sentido, a modelagem do problema e o conhecimento existente sobre ele são muito mais valorizados e contribuem mais efetivamente na sua solução.

4.2. Hibridização da Metaheurística Algoritmo Genético.

Com base no trabalho de Ochi et al. (2006), a estrutura convencional do Algoritmo Genético é modificada, de modo a permitir um processo evolutivo melhor.

a) Estratégia de Geração de uma população inicial

Sugere-se a repetição, k vezes, da etapa de construção de solução viável semelhante ao procedimento usado pela Metaheurística GRASP (Blum et al., 2005), gerando-se uma população de k indivíduos com bom potencial.

b) Estratégia de seleção de indivíduos reprodutores

Com o objetivo de evitar parada prematura em ótimos locais distantes do global, associa-se uma probabilidade maior para a escolha de melhores indivíduos e menor para a escolha de piores indivíduos, numa estratégia semelhante à usada pelo *Simulated Annealing* para fugir de ótimos locais.

c) Estratégia de cruzamento

Nos operadores tradicionais de Algoritmos Genéticos o cruzamento é feito envolvendo dois pais. Sugere-se, aqui, a envolvimento de dois ou mais pais.

d) Incorporação de busca local.

Neste algoritmo evolucionário, propõe-se a incorporação de busca local sobre melhores indivíduos, com o objetivo de explorar sua vizinhança.

5. CONCLUSÃO

O uso de procedimentos híbridos é uma tendência verificada na exploração de espaços de soluções de problemas de natureza combinatória, como pode ser observado nas publicações feitas pela comunidade científica em anais de congressos, periódicos e livros. Por se tratar de um complexo problema de natureza combinatória, o problema de planejamento florestal, representado pela necessidade de estabelecer uma seqüência de atividades a serem executadas numa unidade de produção (talhão), de modo a maximizar o Valor Presente Líquido (VPL) da exploração florestal e atendendo a restrições técnicas ou de mercado, é candidato á aplicação desses procedimentos.

As estratégias de busca sugeridas neste trabalho, por meio de hibridização de Programação por Restrições e Metaheurísticas são promissoras, uma vez que foram aplicadas com êxito em outros tipos de problemas de natureza combinatória.

A geração das prescrições através da ferramenta ZDCDirect utilizando as técnicas de Programação por restrições, mostrou uma alternativa viável para a solução de problemas de planejamento da produção florestal.

O problema teste utilizado, apresenta informações de uma empresa de pequeno, podendo posteriormente ser ampliado para um problema de grande, utilizando recursos computacionais de última geração.

Vale lembrar, no entanto, que o gestor florestal enfrenta situações no seu dia a dia que podem fugir da realidade que o modelo consegue captar, o

que faz com que as soluções oferecidas pelo modelo ao gestor sejam apenas um auxílio no complexo processo de tomada de decisão.

REFERÊNCIAS BIBLIOGRÁFICAS

ALBA, E. (editor), **Parallel Metaheuristics: a new class of Algorithms**. John Wiley & Sons. Hoboken, N.J., 2005.

ALBA, E., et al., Metaheuristics and Parallelism. In: **Parallel Metaheuristics: a new class of Algorithms**. [edited by E. Alba], John Wiley & Sons. Hoboken, N.J., 2005.

BARTÁK, R., **Constraint Programming: In Pursuit of the Holy Grail**. Charles University, Czech Republic, 1999. 10p.

BLUM, C., et al., An Introduction to Metaheuristics Techniques. In: **Parallel Metaheuristics: a new class of Algorithms**. [edited by E. Alba], John Wiley & Sons. Hoboken, N.J., 2005.

BRADWELL, R., et al., **An overview of the CACP project: modelling and solving constraint satisfaction/optimization problems with minimal expert intervention**. Colchester, 2004. Disponível em <<http://cswww.essex.ac.uk/CSP/cacp/CacpOverview2000.htm>>. Acesso em: 20 dez. 2004.

BRAZ, E. M., **Um modelo em programação linear para garantia do rendimento sustentado em pequena propriedade na floresta tropical**. 2001. 76 f. Dissertação (Mestrado em Engenharia Florestal) Universidade Federal do Paraná, Curitiba, 2001.

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

CHOOSE TECHNOLOGIES, **Sistemas de Otimização de Recursos**. Choose Technologies, São Paulo, SP: 2003. 17p.

CIRÉ, A. A. et al., **Hibridização de Programação por Restrições e Metaheurísticas Paralelas para o Escalonamento Integrado de Veículos e Tripulação**. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, XXXVII, 2005, Gramado, RS. Anais do XXXVII SBPO - Simpósio Brasileiro de Pesquisa Operacional, Rio de Janeiro, RJ, 2006.

FATURETO, C.R.C., **Otimização sob critérios múltiplos: metodologias e uma aplicação para o planejamento agrícola**. Tese (Doutorado), Universidade Federal de Viçosa, Viçosa, MG, 1997. 145p.

FIKES, R., **A heuristic program for solving problems stated as non-deterministic procedures**. PhD thesis, Carnegie Mellon University. 1968.

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

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.

JAFFAR, J.; LASSEZ, J. L., **Constraint logic programming**. In **POPL'87: Proceedings** 14th ACM Symposium on Principles of Programming Languages, pp 111-119. 1987.

KIRKPATRICK JR., S., et al., **Optimization by simulated annealing**. Science, v. 220, n. 4598, p. 498-516, 1983.

MARCHIORO, G. B., **O Uso da Programação Lógica por Restrições no Problema de Escalonamento de Tarefas com Restrições de Recursos: Uma Aplicação em Manutenção Industrial**. Monografia (Especialização em Métodos Numéricos em Engenharia) – Universidade Federal do Paraná, Curitiba, PR, 2003.

MARRIOT, K.; STUCKEY, P.J., **Programming with Constraints: An Introduction**. 1.ed. Cambridge, Massachusetts: MIT Press, 1998.

MELLO, A. A., et al., **Planejamento Florestal Visando à Maximização dos Lucros e a Manutenção do Estoque de Carbono**. Cerne, Lavras, v. 11, n. 3, p. 205-217, jul./set. 2005

MILLS, P. et al., **EaCL 1.5: An Easy abstract Constraint optimisation Programming Language**. Technical report CSM-324. Department of Computer Science - University of Essex. Colchester, Essex, United Kingdom. 1999. 15p.

MONTANARI, U., **Network of constraints: Fundamentals properties and applications to picture processing**. Information Science, 7, pp 95-132. 1974.

MÜLLER, T., **Constraint Propagation in Mozart**. PhD thesis, Universität der Saarlandes. Saarbrücken: 2001

NOBRE, S. R., **A heurística da Razão-R aplicada a problemas de gestão florestal**. 1998. 98f. Dissertação (Mestrado), Universidade Federal de Lavras, Lavras, MG, 1999.

OCHI, L. S. e TRINDADE, A. R., **Um algoritmo evolutivo híbrido para a formação de células de manufatura em sistemas de produção**. Pesquisa Operacional, v. 26, n.2, p. 255-294, 2006..

PEREIRA, G. W., **Aplicação da Técnica de Recozimento Simulado em Problemas de Planejamento Florestal Multiobjetivo**. 84f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Minas Gerais, Belo Horizonte, MG, 2004.

RODRIGUES, F. L., **Metaheurística e sistema de suporte à decisão no gerenciamento de recursos florestais**. 2001. 255p. Tese (Doutorado), Universidade Federal de Viçosa, Viçosa, MG, 2001.

RODRIGUES, F. L., **Solução de Problemas de Planejamento Florestal com Restrições de Inteira utilizando Busca Tabu**. In: SIF - Sociedade de Investigações Florestais. Viçosa, MG, 2003. 13p.

RODRIGUEZ, L.C.E. et al., **Técnicas e modelos quantitativos de suporte à decisão florestal** – Programa de reciclagem em métodos quantitativos. ESALQ-USP-IPEF, Piracicaba, 1996. 21p. Apostila.

ROSA, V. B., **Sistema para Logística de Distribuição**. 68f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Universidade Regional de Blumenau, Blumenau, SC, 2005.

RUSSELL, S.; NORVING, P., **Artificial intelligence: a modern approach**. 2 ed. New Jersey: Prentice-Hall, 2003. 1080 p.

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.

STEELE, G. L., **The definition and implementation of a computer programming language based on constraints**. PhD thesis, MIT. 1980.

SUCUPIRA, I. F., **Programação por Propagação de Restrições: Teoria e Aplicações**. Relatório Final de Iniciação Científica, Universidade de São Paulo, São Paulo, SP, 2003. 14p. Disponível em: <<http://www.ime.usp.br/~igorrs/ic/relatorio/relatorio.pdf>>. Acesso em: 20 set. 2004.

SUTHERLAND, I., **Sketchpad: a man-machine graphical communication system**. In Proceedings of the IFIP Spring Joint Computer Conference. 1963.

TAVARES, J. A. R., **Generation of Industrial Systems Configurations by**

Using the Constraint Technology and Evolutionary Computation. Tese (Doutorado em Informática) - University Minho, Guimarães, Portugal, 2000.

TEIXEIRA, A. F., **Aplicação de Algoritmos Evolucionários na Solução de Problemas de Planejamento Florestal Multiobjetivo**, 72f. Dissertação (Mestrado em Ciência da Computação), Universidade Federal de Minas Gerais, Belo Horizonte, MG, 2002.

TSANG, E., **Foundations of constraint satisfaction.** London: Academic Press Limited, 1993. 421 p.

WALTZ, D.L., **Understanding line drawings of scenes with shadows. In: Psychology of Computer Vision**, McGraw-Hill, New York, 1975.