

Programa de Pós-Graduação em Engenharia da Computação

Uma Abordagem Evolucionária Multi-Objetiva para Suporte à Decisão de Colheitas de Cana-de- Açúcar

Dissertação de Mestrado

Engenharia da Computação

Diogo Ferreira Pacheco
Orientador: Prof. Fernando Buarque de Lima Neto

Recife, Junho de 2008.



Uma Abordagem Evolucionária Multi-Objetiva para Suporte à Decisão de Colheitas de Cana-de- Açúcar

Dissertação de Mestrado

Engenharia da Computação

Esta Dissertação é apresentada como requisito parcial para obtenção do título de Mestre em Engenharia da Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Diogo Ferreira Pacheco
Orientador: Prof. Fernando Buarque de Lima Neto

Recife, Junho de 2008.



Diogo Ferreira Pacheco

**Uma Abordagem Evolucionária
Multi-Objetiva para Suporte à
Decisão de Colheitas de Cana-de-
Açúcar**

Resumo

A cana-de-açúcar é uma das *commodities* mais disputadas e com volumes de comercialização de maior expansão no mundo. Esse interesse é motivado por sua importância econômica advinda de seus principais sub-produtos: o açúcar, o álcool e a produção de energia térmica. O Brasil é o maior produtor mundial de cana-de-açúcar, com produção anual maior que a soma dos quatro maiores produtores seguintes.

A colheita da cana-de-açúcar é uma das etapas mais importantes na complexa atividade canavieira. As decisões de colheita têm impacto elevado e direto na produtividade, e conseqüentemente, na rentabilidade do agro-negócio. Talvez essa complexidade explique, em muitas usinas, a realização do planejamento de colheita de forma empírica.

Nesta dissertação registramos o trabalho de pesquisa onde se desenvolveu uma abordagem evolucionária e multi-objetiva (MO) para apoiar decisões de colheita de cana-de-açúcar. Foram considerados concomitantemente interesses agrônômicos e logísticos, inerentes à atividade, ambos, concorrentes entre si. Ainda, cuidou-se de incorporar no modelo proposto, as preferências gerenciais dos tomadores de decisão quando das sugestões de colheita produzidas pelo modelo.

Esta pesquisa foi desenvolvida em três fases, nas quais o complexo problema de planejamento de colheita foi analisado e tratado, a saber: (i) formulação MO do problema; (ii) inclusão de aspectos logísticos em adição aos aspectos agrônômicos, além de melhorias na modelagem do problema (*i.e.* representação dos dados, operadores genéticos e algoritmos utilizados); e finalmente, (iii) tratamento das preferências gerenciais do tomador de decisão.

Em adição ao modelo proposto, construímos um aplicativo inteligente híbrido bastante flexível, que incorpora redes neurais artificiais, algoritmos evolucionários multi-objetivo para otimização da busca pela melhor decisão e lógica fuzzy.

Para validar o modelo, utilizamos dados de três bases reais de usinas do sudeste brasileiro. Nos experimentos realizados foram analisados aspectos MO diversos, sejam eles computacionais (*e.g.* diversidade de soluções, tempo de computação e convergência para o Pareto ótimo) bem como aspectos econômicos (*e.g.* agrônômicos e logísticos).

Os resultados obtidos comprovaram que o modelo proposto e implementado conseguiu (i) aumentar a produtividade das safras de cana-de-açúcar simuladas e (ii) tornar o processo de decisão e planejamento de colheita de cana-de-açúcar menos empírico.

Abstract

Sugarcane is presently one of the most valued commodities with commercialization figures on rapid expansion worldwide. This interest is mainly motivated by economic reasons of its derived products, namely, sugar, ethanol and production of thermal energy. Brazil is the biggest sugarcane producer in the world with annual production larger than the sum of the following four top producers.

The sugarcane harvest is one of the most important tasks in this complex agricultural activity. Harvest decisions have a strong and direct influence on the sugarcane productivity, and consequently, on the agribusiness as a whole. Perhaps, this complexity explains why empirical harvest planning is still performed in many sugarcane plantations.

In this dissertation, we recorded all research work carried out in which an evolutionary and multi-objective (MO) approach was developed to support sugarcane harvest decisions. We have considered in the proposed model the conflicting agronomic and logistic criteria, inherent in the agronomical activity. Harvest suggestions produced by our model, also incorporate management preferences of the decision makers.

This research was performed in three steps, where the complex planning problem was analyzed and tackled in a combined manner, they are: (i) MO problem formulation; (ii) inclusion of logistic aspects in addition to the agronomic ones, as well as several improvements in problem modeling (*i.e.* data representation, genetic operators and used algorithms); and finally, (iii) considerations on how to include the decision maker's management preferences.

In addition to the proposed model, a quite flexible hybrid intelligent application was built, which incorporates artificial neural networks, multi-objective evolutionary algorithms for optimization of search for better decisions, and fuzzy logic.

Three real data sets from sugarcane plantations of Brazilian southeast were used to validate the model. In the experiments carried out in this research several MO aspects were analyzed, whether of computational nature (*e.g.* solutions diversity, run time and optimal Pareto convergence) or of economical nature (*e.g.* agronomics and logistics).

The obtained results confirmed that the proposed approach and implemented model was successful in (i) increasing the simulated sugarcane productivity and (ii) reducing the empiricism in the harvest planning and decision-making process.

Sumário

Índice de Figuras	v
Índice de Tabelas	viii
Tabela de Símbolos e Siglas	ix
Agradecimentos	xi
1 Introdução	12
1.1 Motivação	12
1.2 Caracterização do Problema	13
1.3 Objetivos	15
1.4 Resultados esperados	16
1.5 Estrutura da Dissertação	17
2 Domínio de Técnico	19
2.1 Algoritmos Genéticos	19
2.1.1 Princípios de Trabalho	19
2.1.2 Representando uma Solução	21
2.1.3 Atribuindo Aptidão a um Indivíduo	21
2.1.4 Operador de Seleção	21
2.1.5 Operador de Cruzamento	22
2.1.6 Operador de Mutação	22
2.1.7 Parâmetros de um GA	23
2.2 Otimização Multi-Objetivo Evolucionária	23
2.2.1 Problema de Otimização Multi-Objetivo	24
2.2.2 Princípios da Otimização Multi-Objetivo	25
2.2.3 Busca e Tomada de Decisão	28
2.2.4 Por Que Otimização Evolucionária?	30
2.2.5 A Ascensão dos Algoritmos Evolucionários Multi-Objetivo	32
2.2.6 Principais Algoritmos Multi-Objetivo Usados Atualmente	34
2.2.7 Métricas de Desempenho	38
2.3 Problemas da Mochila	42
2.3.1 História	42
2.3.2 Formulação	43
2.3.3 O Problema da Mochila e os Algoritmos Evolucionários	45
2.4 Redes Neurais Artificiais	47
2.4.1 Definição de RNA	48
2.4.2 Processos de Aprendizagem	49
2.4.3 Redes <i>Multi-Layer Perceptron</i>	50
3 Domínio de Aplicação	56
3.1 Cana-de-açúcar	56
3.1.1 História	56

3.1.2	Botânica e Fisiologia	57
3.1.3	O Processo de Colheita	58
3.1.4	Fatores Determinantes de Produtividade	59
3.1.5	Importância Econômica da Cultura Canavieira	59
3.2	Suporte à Decisão Gerencial	61
3.2.1	Tipos de Sistemas de Informação	62
3.2.2	Tomada de Decisão e Solução de Problemas	62
3.2.3	Visão Geral de Sistemas de Suporte à Decisão	65
3.3	Sistemas de Informação para Cana-de-Açúcar	66
3.3.1	Hybrid Intelligent Suite for Decision Support - HIDS	67
4	Uma Abordagem Evolucionária Multi-Objetiva para Suporte à Decisão de Colheitas de Cana-de-Açúcar	69
4.1	Definição do Problema	69
4.2	Visão Geral do Sistema de Apoio à Decisão	71
4.2.1	Modelo Abstrato para Tomada de Decisão Multi-Objetiva	72
4.2.2	Incluindo Habilidades Multi-Objetivos em HIDS	73
4.3	Evolução da Proposta	74
4.3.1	Fase Um – Formulação Multi-Objetiva	75
4.3.2	Fase Dois – Incorporação de Aspectos Logísticos e Paradigma do Problema da Mochila	77
4.3.3	Fase Três – Meta Aplicação de EMOO	81
5	Experimentos Realizados	87
5.1	Bases de Dados	87
5.1.1	Base CLIENTE_0	88
5.1.2	Base CLIENTE_1	89
5.1.3	Base CLIENTE_2	92
5.2	Modelagem Neural	95
5.2.1	Modelagem neural para CLIENTE_0	95
5.2.2	Modelagem neural para CLIENTE_1 e CLIENTE_2	95
5.3	Resultados da Evolução da Proposta	98
5.3.1	Resultados da Fase Um	98
5.3.2	Resultados da Fase Dois	104
5.3.3	Resultados da Fase Três	115
5.3.4	Resumo dos Resultados	120
6	Conclusões e Trabalhos Futuros	121
6.1	Contribuições	121
6.1.1	Contribuições Fase Um	121
6.1.2	Contribuições Fase Dois	122
6.1.3	Contribuições Fase Três	123
6.2	Discussão	123
6.3	Trabalhos Futuros	124
	Bibliografia	126
	Apêndice A – Trabalhos relacionados com o tema desta dissertação publicados pelo autor	140

Índice de Figuras

Figura 2.1. Fluxograma do princípio de trabalho de um algoritmo genético.	20
Figura 2.2. Representação de um espaço de variáveis de decisão e seu correspondente espaço objetivo.....	25
Figura 2.3. Gráfico formado entre o tamanho da tela e o preço de TVs LCD pesquisadas.	26
Figura 2.4. Esquema de decisão via um procedimento de otimização multi-objetivo baseado em preferências (adaptado de [Deb2001])......	29
Figura 2.5. Esquema de decisão via um procedimento ideal de otimização multi-objetivo (adaptado de [Deb2001])......	29
Figura 2.6. Esquema de decisão via um procedimento iterativo de otimização multi-objetivo (adaptado de [Fonseca1998a])......	30
Figura 2.7. Diagrama de classificação de algoritmos de otimização, adaptado de [Alba2007]. ...	31
Figura 2.8. Crescente número de publicações de estudos em MOEA de 1967 até o início de 2007 (adaptado de [Coello2007b])......	34
Figura 2.9. Pseudo-código do algoritmo SPEA2.	36
Figura 2.10. Pseudo-código do algoritmo NSGA-II.	37
Figura 2.11. Pseudo-código do algoritmo PESA-II.	38
Figura 2.12. Dois objetivos de MOO e um conjunto de soluções não-dominadas ideal em um problema hipotético, adaptado de [Deb2001].	39
Figura 2.13. Um exemplo da representação da métrica hipervolume. Em cinza, a região do espaço de busca explorada pela população.....	41
Figura 2.14. Neurônio artificial MCP desenvolvido por McCulloch e Pitts [McCulloch1943]....	48
Figura 2.15. Diferentes taxa de aprendizado e suas respectivas curvas de aprendizado. Adaptada de [Principe2000]......	53
Figura 2.16. Relação entre taxa de aprendizado e convergência a partir do ajuste de pesos, extraída de [Principe2000].	53
Figura 2.17. MSE de treinamento e validação – critério de parada GL5.	54
Figura 3.1. Lista dos 10 maiores países produtores de cana-de-açúcar-de-açúcar do mundo [FAO2007].	60

Figura 3.2. Evolução da produção de cana-de-açúcar no Brasil entre os anos de 2001 e 2007. Extraído de [MinisterioAgricultura2008a].	61
Figura 3.3. Classificação de sistemas de informação de acordo com O'Brien [O'Brien2001].	62
Figura 3.4. Ontologia de decisão proposta por Oliveira <i>et al.</i> [Oliveira2006]. Cada decisão pode ser decomposta como componentes e seus atributos (<i>i.e.</i> variáveis).	67
Figura 3.5. Visão geral do <i>hybrid intelligent decision suite</i> , adaptado de [LimaNeto2007].	68
Figura 4.1. Nova ontologia de decisão que foca objetivos ao invés de componentes como no original HIDS [LimaNeto2007]. Cada solução viável pode ser decomposta como funções objetivos e suas variáveis de decisão.	72
Figura 4.2. Visão geral do novo MO-HIDS com habilidades de suporte à decisão multi-objetivo.	74
Figura 4.3. Distribuição física hipotética dos lotes de cana-de-açúcar e possíveis definições de vizinhança.	81
Figura 4.4. Sistema de colheita que usa uma meta-aplicação de EMOO para definir as heurísticas que serão utilizadas pelo outro nível de aplicação.	83
Figura 4.5. Funções de pertinência aplicadas na utilização das heurísticas fuzzy.	85
Figura 5.1. Base de dados CLIENTE_1 antes do pré-processamento: atributos e distribuição.	90
Figura 5.2. Base de dados CLIENTE_1 após o pré-processamento.	91
Figura 5.3. Base de dados CLIENTE_2 antes do pré-processamento: atributos e distribuição.	93
Figura 5.4. Base de dados CLIENTE_2 após o pré-processamento.	94
Figura 5.5. Indivíduo que representa a sugestão de colheita de 15 lotes entre os 40 disponíveis.	100
Figura 5.6. Convergência das populações após otimização multi-objetivo nas abordagens: <i>a</i> (A) e com a abordagem <i>b</i> (B).	101
Figura 5.7. Superfícies do Pareto fronte obtidos com a abordagem <i>a</i> (A) e com a abordagem <i>b</i> (B).	102
Figura 5.8. Decodificação de duas soluções encontradas pela abordagem <i>b</i> (blocos da esquerda e central). Cada quadrado colorido indica que o lote foi escolhido para a colheita: na cor azul pela solução I e na cor amarela pela solução II. O bloco mais a direita sobrepõe as duas soluções e os lotes selecionados por ambas aparecem de verde.	103
Figura 5.9. Uma comparação relativa entre a tonelagem de cana-de-açúcar produzida pelas diferentes técnicas de otimização multi-objetivo aplicadas na <i>Fase Um</i> desta pesquisa.	104
Figura 5.10. Hipervolume para as diferentes abordagens com a base CLIENTE_1.	108
Figura 5.11. Hipervolume para as diferentes abordagens com a base CLIENTE_2.	108

Figura 5.12. Espalhamento (GS) para as diferentes abordagens com a base CLIENTE_1.....	110
Figura 5.13. Espalhamento (GS) para as diferentes abordagens com a base CLIENTE_2.....	110
Figura 5.14. Tempos de execução para as diferentes abordagens utilizando a base CLIENTE_1.	111
Figura 5.15. Tempos de execução para as diferentes abordagens utilizando a base CLIENTE_2.	111
Figura 5.16. Espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_1.....	113
Figura 5.17. Curvas de nível de TE no espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_1.	113
Figura 5.18. Espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_2.....	114
Figura 5.19. Curvas de nível de TE no espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_2.	114
Figura 5.20. Comparação multi-objetiva de HV x GS das abordagens simuladas.	117
Figura 5.21. <i>Trade-offs</i> com objetivos agronômicos e logístico obtido com a abordagem <i>5cpPESA</i>	119
Figura 5.22. <i>Trade-offs</i> com apenas os objetivos agronômicos obtidos com a abordagem <i>5cpPESA</i>	120

Índice de Tabelas

Tabela 4.1. Variáveis consideradas no problema da colheita da cana-de-açúcar.	70
Tabela 4.2. Configurações de representação, tratamento de restrição e operadores genéticos adotados do problema da mochila e aplicadas no problema da colheita.	79
Tabela 4.3. Heurísticas <i>fuzzy</i> utilizadas e suas prioridades.	85
Tabela 4.4. Seis exemplos de soluções e suas elegibilidades dentre as heurísticas fuzzy.	86
Tabela 5.1. Definição e exemplo de registro da base de dados CLIENTE_0 (pré-processada). ...	89
Tabela 5.2. Limites inferiores e superiores de atributos nas bases CLIENTE_1 e CLIENTE_2. .	92
Tabela 5.3. Topologia, configuração e MMRE da RNA usada com a base CLIENTE_0.	95
Tabela 5.4 Diferentes configurações de pré-processamento confrontadas nas bases de dados CLIENTE_1 e CLIENTE_2 e as respectivas quantidades de bits necessárias.	97
Tabela 5.5. Melhores resultados de predição e parametrização para diferentes abordagens aplicadas às bases CLIENTE_1 e CLIENTE_2.	98
Tabela 5.6. Melhor configuração paramétrica encontrada para o problema abordado na <i>Fase Um</i>	100
Tabela 5.7. Valores das produções de cana-de-açúcar, PCC e FIBRA alcançadas pelos melhores métodos clássicos e evolucionários aplicados no problema.	104
Tabela 5.8. Parâmetros utilizados em ambas as bases de dados nos experimentos realizados na <i>Fase Dois</i>	106
Tabela 5.9. Configurações para o primeiro e o segundo nível de aplicação EMOO.	116
Tabela 5.10. Abordagens utilizadas nos experimentos da <i>Fase Três</i>	117
Tabela 5.11. Classificação das cinco melhores abordagens entre as 15 pré-definidas e aplicadas em cada um dos cenários hipotéticos.	118
Tabela 5.12. Classificação das abordagens de acordo com o critério de pontuação definido para os cenários hipotéticos.	119

Tabela de Símbolos e Siglas

(Dispostos por ordem alfabética)

ACO	Otimização por colônia de formigas (do inglês, <i>Ant Colony Optimization</i>)
AHP	<i>Analytic Hierarchy Process</i>
ATR	Açúcar Total Recuperável
BKP	<i>Bounded Knapsack Problem</i>
DM	Tomador de decisão (do inglês, <i>Decision Maker</i>)
EA	<i>Evolutionary Algorithms</i>
EMOO	<i>Evolutionary Multi-Objective Optimization</i>
FAO	Organização das Nações Unidas para a Agricultura e Alimentação (do inglês, <i>Food and Agriculture Organization of the United Nations</i>)
FIBRA	Quantidade de material seco resultante de moagem que será aproveitado como potencial calorífico.
FSS	<i>Fish School Search</i>
GA	<i>Genetic Algorithms</i>
GS	<i>Generalized Spread</i>
HIDS	<i>Hybrid Intelligent Decision Suite</i>
HIS	Sistemas inteligentes híbridos (sigla do inglês <i>Hybrid Intelligent Systems</i>)
HV	<i>Hypervolume</i>
HVR	<i>Hypervolume Ratio</i>
Int \$	Média de preços internacionais em dólar
IPP	<i>Integer Programming Problem</i>
KP	Problemas da mochila (do inglês, <i>Knapsack Problems</i>)
MAUT	<i>Multi-Attribute Utility Theory</i>
MChKP	<i>Multiple-Choice Knapsack Problem</i>
MCoKP	<i>Multiple-Constrained Knapsack Problem</i>
MKP	Problema das múltiplas mochilas (do inglês, <i>Multiple Knapsack Problem</i>)
MLP	Múltiplas camadas de Perceptron (do inglês, <i>Multilayer Perceptron</i>)
MMRE	Magnitudes médias de erro relativo obtidas (do inglês, <i>mean magnitude of relative error</i>)
MO	Multi-Objetivo
MOEA	Algoritmo evolucionário multi-objetivo (do inglês, <i>Multi-Objective Evolutionary Algorithm</i>)
MOGA	<i>Multi-Objective Genetic Algorithm</i>
MOO	<i>Multi-Objective Optimization</i>
MOOP	<i>Multi-Objective Optimization Problem</i>
MSE	Erro médio quadrático (do inglês, <i>Mean Square Error</i>)
NPGA	<i>Niched Pareto Genetic Algorithm</i>
NSGA	<i>Non-Dominated Sorting Genetic Algorithm</i>

NSGA-II	<i>Elitist Non-Dominated Sorting Genetic Algorithm</i>
PAES	<i>Pareto Archived Evolution Strategy</i>
PCC	Porcentagem aparente de sacarose no caldo da cana-de-açúcar
PESA-II	<i>Pareto Envelop-based Selection Algorithm-II</i>
PSO	<i>Particle Swarm Optimization</i>
RNA	Rede Neural Artificial
SI	Sistemas de Informação
SIA	Sistemas Imunológicos Artificiais
SIG	Sistemas de Informação Gerencial
SNNS	<i>Stuttgart Neural Network Simulator</i>
SOM	<i>Self Organized Maps</i>
SOO	<i>Single-Objective Optimization</i>
SOOP	<i>Single-Objective Optimization Problem</i>
SPEA2	<i>Strength Pareto Evolutionary Algorithm 2</i>
SSD	Sistemas de Suporte à Decisão
TCH	Tonelagem de Cana por Hectare
TD	Tonelagem Desejada
TL	Tonelagem liberada
UKP	<i>Unbounded Knapsack Problem</i>
VEGA	<i>Vector Evaluated Genetic Algorithm</i>
μ GA2	<i>Micro Genetic Algorithm</i>

Agradecimentos

Agradeço a Deus pelo dom da vida. Agradeço a chance de hoje poder fazer parte de uma minúscula parte da população que teve direito a uma formação superior e a uma pós-graduação. Peço que todas essas graças recebidas possam de alguma forma ser retribuídas para a formação de uma sociedade mais igualitária, justa e humana.

Agradeço a Deus pela minha amada família, que é a minha maior herança – o melhor presente que Deus me deu. Mainha (Geyne), painho (Pacheco) e Igor (irmão) essa conquista é tão de vocês quanto minha. Muito obrigado pelo apoio, incentivo e principalmente paciência. Não posso deixar de mencionar as palavras de estímulo de minhas avós, tias (os) e primas (os).

Agradeço a Deus pela minha namorada e companheira Erika, que diretamente sofreu com minhas angústias, meus receios, meus prazos e principalmente com meu mau-humor estressado. Sua doação, zelo e amor foram, indubitavelmente, energizantes para que eu pudesse continuar na batalha.

Agradeço a Deus pelos meus amigos irmãos (alagoanos e pernambucanos) que também foram exemplos de perseverança, determinação e cumplicidade, nos estudos e no trabalho. Não citarei nomes, pois certamente esqueceria de alguém, mas destaco os companheiros de batalha, alunos desta primeira turma de mestrado da Escola Politécnica de Pernambuco, que foram firmes nas brigas, unidos nas dificuldades e que agora também compartilham desta conquista.

Agradeço a Deus pelos excelentes professores que tive, os quais me proporcionaram uma formação de excelência. Agradeço também a dedicação e a preocupação de Georgina, que foi secretária formalmente, mas que na realidade foi uma mãe para todos nós. Um agradecimento especial a Fernando Buarque que foi muito mais que um professor e um orientador, mas foi um amigo e educador para toda a vida. Sem sombras de dúvidas, foi a pessoa que mais exigiu, apoiou, reanimou e acreditou na minha formação. Agradeço também a Mônica, que gentilmente cedeu várias horas de Fernando nos fins de semana em prol de seus orientados.

Gostaria ainda de pedir desculpas a Deus pelas minhas fraquezas e pouca fé, tão facilmente abaladas.

Enfim, a todos vocês, meu **MUITO OBRIGADO!**

Capítulo 1

Introdução

1.1 Motivação

Atualmente, um dos assuntos que vêm gerando grandes debates internacionais é a escassez de alimentos. As elevações dos preços do trigo, milho, soja, arroz, entre outros são as maiores já observadas nos últimos anos. As conseqüências imediatas são: aumento nos preços, bloqueio de exportações, medidas para conter o consumo [Khor2007]. A Organização das Nações Unidas para a Agricultura e Alimentação (do inglês, *Food and Agriculture Organization of the United Nations*, FAO) argumenta que o crescente uso de alimentos (cereais, grãos, açúcar e oleaginosas) para a geração de biocombustíveis é um crime contra a humanidade (que morre de fome), no entanto, como afirma o ministro brasileiro para assuntos do Desenvolvimento Social e Combate à Fome, Patrus Ananias, esses embates entre alimentos e biocombustíveis não se aplicam à realidade brasileira devido à nossa imensa extensão territorial que pode viabilizar tanto a produção alimentícia como de combustíveis renováveis [Lourenço2008].

As altas nos preços dos alimentos podem ser justificadas (i) com safras sendo arrasadas por secas ou cheias causadas pelo aquecimento global; (ii) pela crise no petróleo que aumenta os custos de produção; e (iii) pelas vastas áreas de cultivo, antes destinadas a alimentos, e que agora se destinam à produção de biocombustível [Khor2007]. Do outro lado temos o desenfreado aquecimento global, um vilão que está produzindo mudanças cruciais no mundo, mas não apenas no clima o que compromete o futuro da Terra e, conseqüentemente, a existência da raça humana [Llewellyn2007]. Essas mudanças estão induzindo transformações comportamentais, sociais e econômicas [Gore2006].

Já o aumento de interesse pelos biocombustíveis é motivado, principalmente, (i) pelo próprio aquecimento global e (ii) pela crise mundial do petróleo. Assim, os combustíveis renováveis podem diminuir o aquecimento global e são alternativas para o petróleo (que influencia nos preços dos alimentos) desde que suas produções sejam equilibradas.

A cana-de-açúcar, de acordo com a FAO, é uma das mais importantes *commodities* do mundo [FAO2007]. Seus principais produtos são o açúcar e o álcool, sendo este considerado o melhor combustível renovável [Investing2007]. Não é preciso ressaltar a relevância destes produtos em minorar problemas sociais como: a fome e o aquecimento global. Brasil¹, Índia, China e Tailândia são os quatro maiores produtores de cana-de-açúcar no mundo. Assim como também são populosos países emergentes, podem ser considerados causa e solução para estes problemas. Desta maneira, estudos que objetivam aumentar a produtividade desta *commodity*, devem ser obviamente encorajados e muito bem recebidos.

Neste cenário, com a crescente demanda mundial, por alimento e energia, a produção e a produtividade do cultivo da cana-de-açúcar devem ser aumentadas. Com a magnitude do volume da produção brasileira, até mesmo mínimos acréscimos percentuais na produção representam milhões de dólares de lucro. Por exemplo, 1% de aumento na produção de 2005 teria sido equivalente a Int² \$87 milhões a mais de lucro.

1.2 Caracterização do Problema

Existem muitos fatores e atividades que influenciam a produção de açúcar e álcool em uma usina. A colheita da cana-de-açúcar, por exemplo, além de contribuir sensivelmente na produção final, é uma das atividades mais complexas de se gerenciar. A cana colhida nos diversos lotes nas fazendas da usina deve ser transportada utilizando caminhões até às mesmas. Neste contexto, saber o melhor período para a colheita de cada lote de cana é um problema complexo. De volta a unidade de moagem, o processamento do vegetal resulta em caldo de cana – que é utilizado na fabricação de açúcar e álcool; e em bagaço – que serve principalmente como fonte energética para a própria usina [Pacheco2006].

Durante cada dia de colheita, os gerentes agrícolas devem escolher quais lotes de cana-de-açúcar serão colhidos para prover uma tonelagem (volume) suficiente para que o processo de

¹ O Brasil é responsável por quase de 36% da produção mundial. Ele sozinho consegue produzir mais que os outros 4 maiores produtores juntos [FAO2007].

² Int \$ representa a média de preços internacionais entre 1999 e 2001 [FAO2007].

moagem seja contínuo. Estas decisões são bastante complexas, pois não há disponibilidade de dados de produção e ainda existem vários objetivos conflitantes para balancear. Como por exemplo, o processo de tomada de decisão deve considerar entre outros fatores: a qualidade do vegetal; o número de frentes de corte (mão-de-obra) e frota de caminhões disponíveis; e a capacidade de moagem da usina. Isto significa que inúmeros objetivos agrônômicos intrínsecos devem ser maximizados ao mesmo tempo minimizando os custos logísticos associados à atividade econômica [Alexander1973]. Assim, define-se um problema combinatorial NP-completo [Papadimitriou1982].

Tendo em vista a necessidade de uma avaliação econômica das decisões de colheita, convencionou-se o uso de quatro indicadores de produtividade:

- PCC: mede a porcentagem aparente de sacarose (açúcar) no caldo de cana após a moagem. Este é um indicador de qualidade do vegetal colhido;
- ATR: açúcar total recuperável que indica a quantidade de sacarose contida na cana e que é usada para fabricação de açúcar e/ou álcool. Outro indicador de qualidade;
- FIBRA: mede a potência calorífica no bagaço, matéria fibrosa seca após a extração do caldo de cana. Bons indicadores de fibra possibilitam a usina ser auto-suficiente e fornecedora de energia. A fibra também é fonte de produtos secundários, tais como: ração para animais e matéria-prima para manufatura de papel. Este também é um indicador de qualidade do vegetal;
- TCH: mede a tonelagem de cana por hectare. Este é um indicador que garante os volumes de produção, mede a produtividade.

Além das questões fisiológicas da cana-de-açúcar, existe uma lista extensa de outros fatores que influenciam a produtividade de uma safra. Em seguida alguns deles são citados [LimaNeto1998]:

- Fatores comuns a todos os lotes - variedade da cana, número de cortes, data do plantio, data do corte, topografia e tipo do solo;
- Fatores eventuais a alguns lotes - compactação do solo, déficit hídrico, adubação, irrigação, maturação artificial, pragas e incêndios;
- Fatores climáticos - pluviometria, temperatura, umidade do ar, velocidade do vento e evaporação;

- Fatores logísticos - área do lote, distância do lote para a usina, capacidade das frentes de corte, capacidade e tempo médio de transporte, tonelagem mínima processada diariamente para que a atividade seja ininterrupta.

Não é incomum encontrar usinas que realizam a tomada de decisão de colheita de maneira empírica ou heurística devido à quantidade de fatores envolvidos, à difícil correlação entre eles ou a indisponibilidade de dados. Nesses casos, a experiência dos gerentes é intensamente requisitada para lidar com a situação, onde usualmente estes fazem considerações triviais nos dados de produção, tais como: (i) conhecer a data de plantio do lote e (ii) os comportamentos agronômicos gerais das espécies cultivadas.

A proposta inicial de Lima Neto (1997-1998) correlacionou dados históricos de safras para estimar os indicadores agronômicos da cana-de-açúcar usando uma Rede Neural Artificial (RNA) *Multilayer Perceptron* (MLP) [LimaNeto1997][LimaNeto1998]. Este conhecimento gerado a priori agrega mais informação às decisões, e estas podem ser tomadas com mais pertinência. A partir daí, diversos artigos foram produzidos melhorando o módulo de predições ou agregando módulos de buscas que utilizavam os indicadores estimados para sugerir decisões de colheita [Pacheco2005] [Trigo2005] [Pacheco2006] [Alencar2006] [Oliveira2006].

Apesar do problema da colheita da cana-de-açúcar ser multi-objetivo (MO), pela própria definição, nenhum destes trabalhos utilizou técnicas MO em suas formulações ou consideraram qualquer objetivo logístico relacionado à atividade. Assim, identificamos que o problema em questão exige uma abordagem mais abrangente (*i.e.* multi-objetiva considerando logística) para que seus resultados sejam mais efetivos e verossímeis. Identificamos ainda, diversas similaridades entre o problema da colheita de cana-de-açúcar e o problema das múltiplas mochilas (do inglês, *multiple knapsack problem*, MKP) [Pisinger1995]. Desta forma utilizamos trabalhos mais genéricos sobre MKP como inspiração na resolução deste problema específico.

1.3 Objetivos

O objetivo deste projeto é desenvolver uma solução evolucionária multi-objetiva que apóie as decisões de colheita de cana-de-açúcar. Estas decisões devem ser justas com relação a todos os objetivos de colheita, inclusive os logísticos. Além disso, devem atender às restrições do problema e devem incorporar às preferências do tomador de decisão.

Para alcançarmos o objetivo deste trabalho, foram definidas as seguintes metas:

- a) Estender o modelo proposto anteriormente [**Pacheco2006**] para incorporar técnicas multi-objetivas;
- b) Ajustar a formulação e definir como os dados logísticos serão incluídos no problema, aproximando-se ainda mais do problema real;
- c) Realizar a modelagem neural e simulações necessárias para que o módulo preditivo estime os indicadores de produtividade PCC, ATR, TCH e FIBRA - indicadores estes que serão utilizados pelo módulo de decisão multi-objetivo;
- d) Definir matematicamente o problema da colheita de cana-de-açúcar como um problema do tipo múltiplas mochilas;
- e) Aplicar e comparar o desempenho dos algoritmos MO clássicos: AHP, MAUT, ELECTRE, ϵ -Constraint e Lexicográfico; e os algoritmos evolucionários: VEGA, NSGA-II, SPEA2 e PESA-II;
- f) Analisar possíveis melhorias na representação do problema, nos operadores genéticos e na combinação dos algoritmos utilizados;
- g) Definir a estrutura do algoritmo que melhor se aplique neste problema incorporando as simulações e resultados obtidos que a comprovem.

1.4 Resultados esperados

Ao final deste trabalho, espera-se a criação de um algoritmo evolucionário multi-objetivo (do inglês, *multi-objective evolutionary algorithm*, MOEA), implementado um protótipo de uma ferramenta inteligente híbrida, usando RNA e MOEA que proverá suporte aos gestores, de usinas sucro-alcooleiras, nas decisões de colheita da cana-de-açúcar.

O usuário será capaz de obter predições de, pelo menos, quatro indicadores de produtividade para colheita de cana-de-açúcar (*i.e.* PCC, FIBRA, ATR e TCH). Esses indicadores juntamente com informações logísticas (distância física entre os lotes indicados para o corte, proximidade com as unidades de moagem, área, entre outros) serão submetidos a uma otimização multi-objetivo. O otimizador escolhido será resultado de reflexões e experimentos, caracterizando-se por sua melhor aplicabilidade no problema em lide.

Espera-se uma boa precisão nas predições das RNA, uma boa aproximação do Pareto *front* e uma boa diversidade nas soluções sugeridas para a colheita. Desta forma, o diálogo de decisão oferecido pela ferramenta pode conduzir as análises e conseqüentes decisões dos

gerentes. Portanto, o sistema será concebido para ser um facilitador importante na escolha dos lotes que participarão da colheita.

Espera-se ainda que o desenvolvimento desta ferramenta possa apoiar o gerente agrícola na decisão sobre o planejamento de quais lotes devem ser colhidos diariamente, facilitando a gerência do negócio e/ou aumentando a produtividade, especialmente em regiões onde a cultura da cana-de-açúcar seja de importância econômica preponderante.

Categorizando os benefícios podemos ainda citar:

- a) Financeiros e Econômicos: melhoria nos valores obtidos de FIBRA, ATR e PCC da cana colhida e aumento da produtividade. Isto pode representar ganhos significativos dados os elevados volumes financeiros produzidos por esta atividade econômica;
- b) Sociais: possibilidade de gerar melhorias nas condições dos trabalhadores rurais nesta atividade econômica, motivada pelo provável aumento na lucratividade citada no item anterior;
- c) Acadêmicos: pesquisa de novos conhecimentos científicos sobre aplicação de RNA em agricultura e investigação híbrida de algoritmos multi-objetivo evolucionários;
- d) Técnicos: possibilidade de aplicação de um novo MOEA e de uma ferramenta que utiliza RNA e EMOO para suporte à decisão em colheitas de cana-de-açúcar;
- e) Ambientais: com o aumento na produtividade, há a possibilidade de uma maior racionalidade de utilização de recursos ambientais ajudando a conter o aquecimento global.

1.5 Estrutura da Dissertação

Este documento está organizado em seis capítulos, com conteúdos dispostos da seguinte maneira:

No Capítulo 2, descreve-se o domínio técnico aplicado ao longo da dissertação. Detalham-se os fundamentos de algoritmos genéticos, de otimização multi-objetivo evolucionária, do paradigma do problema da mochila e das redes neurais artificiais. O leitor que já está familiarizado com estes assuntos pode seguir para os demais capítulos.

No Capítulo 3, descreve-se o domínio de aplicação desta dissertação. Nele, detalha-se a cana-de-açúcar desde suas características agrônômicas até aspectos de sua exploração econômica. Os fundamentos dos sistemas de suporte à decisão também são brevemente descritos.

O detalhamento da contribuição desta pesquisa é apresentado no Capítulo 4. Nele mostra-se como o problema foi definido matematicamente e como a formulação MO foi desenvolvida. A

evolução da proposta foi dividida em fases, nas quais podemos encontrar as comparações com os algoritmos clássicos, a inclusão de aspectos logísticos e o estudo sobre preferências.

O Capítulo 5 apresenta os experimentos realizados. Nele encontramos também as informações sobre as bases de dados utilizadas, os resultados do pré-processamento e como foi realizada a modelagem neural do módulo preditivo. Os resultados dos experimentos são apresentados em fases, da mesma forma em que foi detalhada a contribuição.

Finalmente, no Capítulo 6, são apresentadas as conclusões. Discute-se a importância dos resultados encontrados e apresentam-se algumas linhas de investigações futuras.

No Apêndice A estão incluídos os resumos dos trabalhos científicos publicados pelo autor que são relacionados com o tema desta pesquisa.

Capítulo 2

Domínio de Técnico

Neste capítulo descrevemos o conjunto de conceitos técnicos aplicados no desenvolvimento deste trabalho. Inicialmente são apresentados os algoritmos genéticos, seguidos por fundamentos e técnicas sobre otimização multi-objetivo. Em seguida são apresentados os conceitos relacionados com o problema da mochila e redes neurais artificiais.

2.1 Algoritmos Genéticos

Nos últimos anos, os algoritmos genéticos (do inglês, *genetic algorithms*, GA) têm sido utilizados como ferramentas de busca e otimização nos mais variados domínios [Deb2001]. As principais razões para seu sucesso são (i) sua vasta aplicabilidade, (ii) facilidade de utilização e (iii) sua perspectiva global [Goldberg1989].

2.1.1 Princípios de Trabalho

Os conceitos básicos dos GA foram inicialmente propostos por John Holland, inspirado nas teorias da evolução e seleção natural do naturalista e fisiologista inglês Charles Darwin [Holland1975]. Algumas idéias fundamentais da genética são artificialmente adaptadas para construir algoritmos de buscas globais que são robustos e que necessitam de poucas informações sobre os problemas a serem resolvidos.

Darwin defendeu que na natureza apenas os indivíduos mais adaptados ao ambiente sobreviviam, *i.e.* conseguiam perpetuar (transmitir) seu código genético ao longo das gerações a partir de seus descendentes – indivíduos gerados através de combinações genéticas com outros indivíduos (cruzamentos). Aqueles menos adaptados possuíam menores probabilidades de

cruzarem e, conseqüentemente, eram extintos da população por um processo de seleção natural. Além da evolução proporcionada pelas recombinações de código genético, Darwin acreditava que existia um outro mecanismo responsável pelos processos evolutivos – a mutação. A mutação é um processo que ocorre com uma freqüência bem menor que os cruzamentos e consiste de pequenas alterações no código genético de um indivíduo. Essas alterações proporcionam mais diversidade entre os indivíduos: se as mudanças introduzidas fossem positivas, os cruzamentos futuros incorporariam as melhorias nas próximas gerações; caso contrário, o indivíduo mutante seria descartado pela seleção natural [Darwin1989].

Holland e Goldberg aplicaram esses conceitos para criar os GA. Cada indivíduo na população representa uma possível solução para um problema proposto. E cada solução (indivíduo) apresentava um nível de adaptabilidade (do inglês *fitness*) ao problema, *i.e.* indivíduos com melhores *fitness* representam melhores soluções para o problema em questão. A Figura 2.1 mostra o fluxograma geral de trabalho de um GA. Os passos são os seguintes:

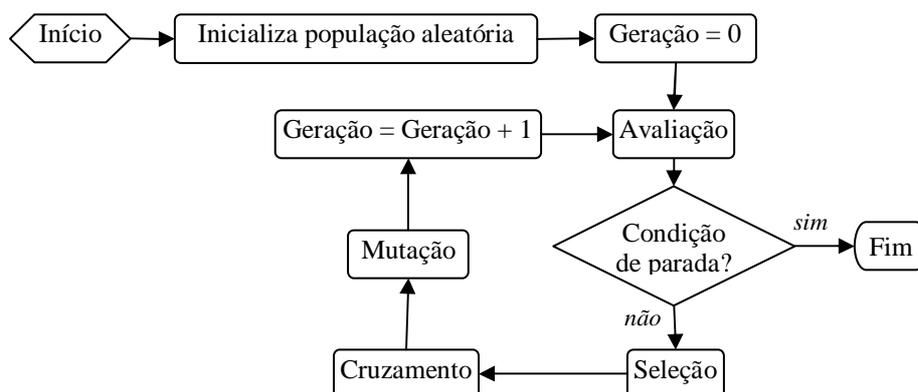


Figura 2.1. Fluxograma do princípio de trabalho de um algoritmo genético.

- i. O processo é iniciado a partir da geração aleatória de uma população inicial;
- ii. Os cromossomos desses indivíduos são decodificados e avaliados para atribuição do *fitness*;
- iii. Verifica se as condições de parada foram atingidas, tais como: solução encontrada, número máximo de gerações, entre outras. Em caso positivo o processo é finalizado, senão repete o ciclo;
- iv. Realiza a seleção dos indivíduos que participarão dos cruzamentos, *i.e.* serão os pais da próxima geração. A seleção deve priorizar os indivíduos com melhores *fitness*;
- v. Cria-se uma nova população a partir dos cruzamentos com os pais selecionados;
- vi. Aplica-se mutação em parte da população garantindo a diversidade e contornando mínimos locais;

vii. Retorna ao passo ii.

2.1.2 Representando uma Solução

Cada indivíduo num GA equivale a uma solução para o problema proposto, assim, modelar o problema (*i.e.* definir a representação de um indivíduo) é uma das atividades mais importantes quando se trabalha com esta técnica. Comumente, nos problemas de busca e otimização, deseja-se encontrar as ótimas variáveis de decisão que satisfaçam as restrições do problema e minimizem / maximizem a função objetivo em questão. A forma mais usual de representar uma solução é com uma string de bits, mas os genes dos indivíduos podem também conter números inteiros, ponto flutuante e até mesmo objetos [Deb2001].

2.1.3 Atribuindo Aptidão a um Indivíduo

Após a modelagem do problema, indivíduos representam soluções para o problema. O primeiro passo na determinação de aptidão (*i.e. fitness*) de um indivíduo é a decodificação do genótipo em fenótipo (*e.g.* uma string com 10 bits na verdade pode representar as variáveis de um problema de engenharia, $h = 3$ e $d = 28$). Avaliar um indivíduo (*i.e.* uma solução) significa calcular o valor da função objetivo e as violações de restrição do problema observado. Quando o problema não apresenta restrições, é comum observar que o *fitness* de uma solução é igual ao valor da sua função objetivo. Por outro lado, na presença de restrições, o *fitness* deve representar uma métrica relativa entre o valor da função objetivo e a violação na restrição. Soluções inviáveis (do inglês, *infeasible*), aqueles que violam alguma restrição do problema, são geralmente penalizadas e apresentam um *fitness* pior que as soluções viáveis (do inglês, *feasible*) [Eiben2003].

2.1.4 Operador de Seleção

O principal objetivo do operador de seleção é escolher os indivíduos que serão responsáveis pela formação da próxima geração, *i.e.* fazer cópias das boas soluções e eliminar as soluções mais fracas da população, mantendo o tamanho da população constante. Para isso, realizam-se os seguintes passos [Deb2001]:

- i. Identificar boas soluções na população, geralmente acima da média;
- ii. Criar múltiplas cópias das boas soluções;
- iii. Eliminar as soluções mais fracas de modo que as cópias das boas soluções criadas ocupem seus lugares na população.

Existem diversos métodos para realizar os passos acima; os mais conhecidos são: torneio, seleção proporcional e ranking [Deb2001] [Eiben2003]. A forma de atribuição do *fitness* é fundamental na identificação das boas soluções. Dependendo da distribuição dos seus valores na população alguns métodos podem ser ineficazes no processo seletivo. Goldberg e Deb mostraram que o torneio de seleção apresenta convergência e custo computacional igual, ou melhor, que qualquer outro método de seleção [Goldberg1991].

Em um dos trabalhos pioneiros sobre GA, DeJong propôs um operador relacionado com a seleção, mas responsável pela preservação dos melhores indivíduos – o elitismo [DeJong1975]. O elitismo garante que os operadores de cruzamento e mutação não eliminarão as melhores soluções encontradas durante o processo de evolução. Basicamente, existem duas formas de realizar o elitismo [Deb2001]: (i) garantindo que a melhor solução será sempre copiada nas novas populações ou (ii) durante a seleção, fazendo a união da população filha com a população anterior ($N + N = 2N$) e escolhendo apenas os N melhores.

2.1.5 Operador de Cruzamento

O operador de cruzamento será aplicado ao grupo selecionado pelo operador de seleção. Note que a seleção não cria indivíduos novos, apenas cópias dos melhores representantes. Os operadores de cruzamento e mutação são os responsáveis pela criação de novas soluções. Em geral, o cruzamento é a combinação dos cromossomos de dois indivíduos (identificados como pais) na geração de duas novas soluções (identificadas como filhas). Os filhos criados não necessariamente são melhores que os pais, mas as chances disto ocorrer são bem maiores que a partir de uma criação aleatória. Assim, ao longo das gerações, a população do GA consegue convergir ao seu objetivo [Deb2001]. Portanto, o operador de cruzamento é responsável pela exploração em profundidade do espaço de busca (em inglês, *exploitation*).

Nem todos os indivíduos da nova população são gerados a partir de cruzamentos, parte dos indivíduos escolhidos durante a seleção permanecem inalterados na próxima geração. A taxa de cruzamento regula a quantidade de combinações e cópias entre gerações. Assim como na seleção, existem diversos métodos de cruzamento e estes estão diretamente relacionados à forma de representação dos indivíduos [Eiben2003].

2.1.6 Operador de Mutação

Com a nova população criada, a partir de cópias e cruzamentos, aplica-se a mutação. Cada indivíduo tem uma probabilidade mínima de sofrer mutação a qual é determinada pela taxa de

mutação. A mutação ocorre individualmente através de pequenas alterações no código genético. O operador de mutação é responsável por uma exploração em largura (*i.e.* aleatória) do espaço de busca (em inglês, *exploration*). Assim, a mutação garante a diversidade entre os indivíduos e permite que durante o processo evolutivo as otimizações não estagnem em mínimos locais. Assim como os operadores de cruzamento, há diversos mecanismos de mutação os quais estão relacionados com a representação do problema [Eiben2003].

2.1.7 Parâmetros de um GA

Além da taxa de cruzamento e da taxa de mutação, existem outros parâmetros que precisam ser ajustados em um GA. Os principais são: tamanho da população e o critério de parada.

A taxa de cruzamento determina quanto da nova população será gerado a partir de cruzamentos ou de cópias de indivíduos antigos. A taxa de mutação, por sua vez, determina o *trade-off* entre *exploration* e *exploitation*, ou seja, determina a parcela de busca aleatória realizada pelo GA. Para determinar o tamanho da população, deve-se estimar a complexidade do espaço de busca. Uma população muito pequena pode implicar em pouco espaço explorado e não convergência; em contrapartida, uma população muito grande pode gerar uma elevada troca de informações (*i.e.* cruzamentos) desnecessárias comprometendo o desempenho do algoritmo. Finalmente o critério de parada, que está intimamente relacionado com a complexidade do problema e, conseqüentemente, com o tamanho da população. Algumas abordagens são utilizadas para sua determinação [Eiben2003]:

- i. Número máximo de gerações;
- ii. Número máximo de avaliações da função objetivo;
- iii. Convergência real: o GA pára quando alcança o ótimo previamente conhecido ou uma solução com erro pré-definido;
- iv. Estagnação: quando não há mudanças significativas no indivíduo de melhor *fitness* durante um número determinado de gerações.

2.2 Otimização Multi-Objetivo Evolucionária

Como o nome sugere, um problema de otimização multi-objetivo (do inglês, *multi-objective optimization problem*, MOOP) inclui mais de uma função objetivo. Na maioria dos problemas práticos de tomada de decisão a existência de vários objetivos ou critérios é bastante freqüente. No passado, diante da carência de metodologias de solução eficientes, os MOOP eram adaptados

e solucionados como problemas de otimização com um único objetivo (do inglês, *single-objective optimization problem*, SOOP). Entretanto, existem inúmeras diferenças fundamentais entre os princípios de trabalho dos algoritmos de otimização de único objetivo e os multi-objetivo. Em SOOP, a missão é encontrar uma solução que otimiza uma função objetivo. Ampliar a idéia para otimização multi-objetivo (sigla do inglês, MOO) pode sugerir erroneamente que a nova missão seria encontrar uma solução ótima para cada função objetivo. Nesta seção são discutidos os princípios de MOO e apresentadas as condições para que soluções possam ser definidas como ótimas quando problemas com múltiplos objetivos estiverem sendo tratados [Deb2001].

2.2.1 Problema de Otimização Multi-Objetivo

Um problema de otimização multi-objetivo possui diversas funções objetivo que precisam ser minimizadas ou maximizadas simultaneamente. Assim como nos problemas de otimização com um único objetivo, também neste caso existem várias restrições as quais qualquer solução viável deve satisfazer. Na seqüência, formula-se o MOOP em sua forma geral:

$$\left. \begin{array}{ll} \text{Minimizar / Maximizar } f_m(\vec{x}), & m = 1, 2, \dots, M; \\ \text{restrito à } g_j(\vec{x}) \geq 0, & j = 1, 2, \dots, J; \\ h_k(\vec{x}) = 0, & k = 1, 2, \dots, K; \\ x_i^{(L)} \leq x_i \leq x_i^{(U)}, & i = 1, 2, \dots, n; \end{array} \right\} \quad (1)$$

A solução \vec{x} é um vetor de n variáveis de decisão: $x = (x_1, x_2, \dots, x_n)^T$. O último conjunto de restrição é chamado de limites das variáveis e restringe cada variável de decisão x_i de assumir valores abaixo $x_i^{(L)}$ e acima $x_i^{(U)}$ dos limites (do inglês, *Lower* e *Upper bounds*, respectivamente limites inferior e superior). Esses limites constituem o espaço de variáveis de decisão D , ou simplesmente, o espaço de decisão.

Associadas ao problema existem possíveis J e K restrições. Os termos $g_j(\vec{x})$ e $h_k(\vec{x})$ são chamados de funções restritivas. Estas restrições, podem ser dos tipos “maior ou igual que” (J), e do tipo “menor ou igual que”, desde que estas sejam multiplicadas por -1 [Deb1995]. Uma solução \vec{x} que não satisfaça a todas as restrições ($J + K$) e a todos os $2N$ limites de variáveis determinados no problema é conhecida como uma solução inviável (do inglês, *infeasible*). Por outro lado se \vec{x} satisfaz a todos esses limites e restrições, a solução \vec{x} é dita viável (do inglês, *feasible*). Portanto, ao serem incluídas restrições, nem todo o espaço de decisão será viável. O

conjunto formado por todas as soluções viáveis de um problema multi-objetivo é chamado de região viável, ou simplesmente, espaço de busca.

A formulação apresentada acima considera M funções objetivos $f(x) = (f_1(x), f_2(x), \dots, f_M(x))^T$. Cada uma delas pode tanto ser minimizada como maximizada. A partir da aplicação do princípio da dualidade³ [Deb1995] [Reklaitis1983] [Rao1984] várias otimizações, que misturam ambos tipos de objetivos, tornam-se mais simples de se resolver. Isto porque existem algoritmos que lidam apenas com maximizações ou minimizações. Para tal, basta multiplicar a função objetivo por -1 quando for o caso. Todas essas M funções criam um espaço multidimensional além do intuitivo espaço de decisão, sendo esta é a principal diferença entre MOO e SOO. Este espaço adicional é conhecido como espaço objetivo Z .

Cada ponto x (solução) no espaço de decisão corresponde a um ponto no espaço objetivo, denotado por $f(x) = z = (z_1, z_2, \dots, z_M)^T$. Este mapeamento, que nem sempre é linear, ocorre entre um vetor de soluções n -dimensional e um vetor de objetivos M -dimensional. A Figura 2.2 exemplifica um mapeamento entre esses dois espaços e o mapeamento entre eles.

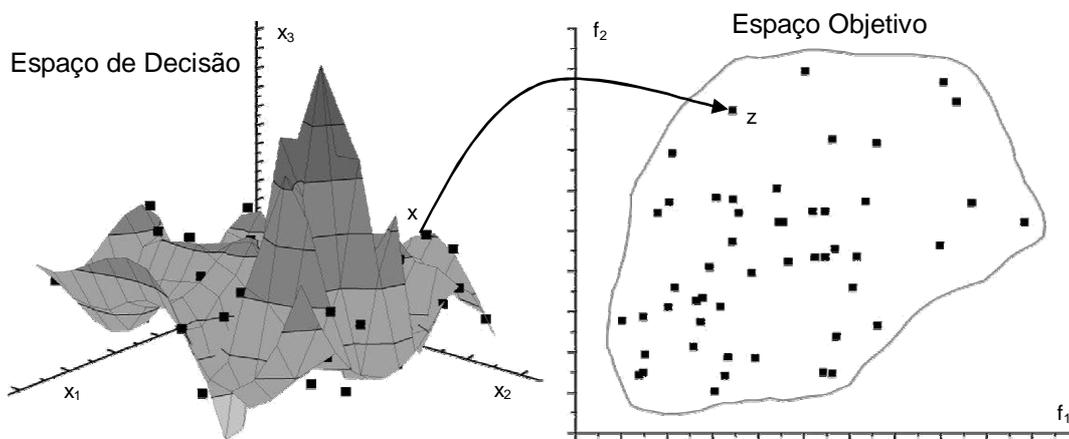


Figura 2.2. Representação de um espaço de variáveis de decisão e seu correspondente espaço objetivo.

2.2.2 Princípios da Otimização Multi-Objetivo

Para ilustrar os princípios de otimização multi-objetivo utilizaremos um exemplo bastante simples, a saber. Com o advento da TV digital a procura por televisores de plasma e LCD aumentou significativamente. Suponha que após ler diversas matérias um consumidor decide que irá adquirir uma TV LCD. Para decidir qual televisor comprar, ele pode considerar uma série de

³ O princípio da dualidade permite converter um problema de minimização em um problema de maximização e vice-versa.

fatores, tais como: marca, preço, resolução, tamanho da tela, tempo de garantia, dentre outros. Se ele decidisse considerar todos estes fatores perderia muito tempo e, provavelmente, não encontraria melhor opção. Assim, conhecendo suas limitações, decide considerar apenas o tamanho da tela e o preço. Em sua pesquisa ele coletou o preço e o tamanho de diversos modelos buscando um que lhe oferecesse o menor preço e a maior tela. Para facilitar sua decisão e descartar modelos indesejados, ele faz algumas reflexões sobre os seus anseios:

- a) Como a TV será colocada a 3m de distância dos usuários, então televisores menores que 15” e maiores que 45” não seriam apropriados ao ambiente;
- b) De acordo com o seu orçamento, a disponibilidade financeira para a compra é de até R\$6.000,00.

Ao final da pesquisa de preços, o comprador do exemplo gerou um gráfico como o da Figura 2.3 com todos os modelos encontrados. Dele, pode-se extrair vários conceitos relacionados com MOO:

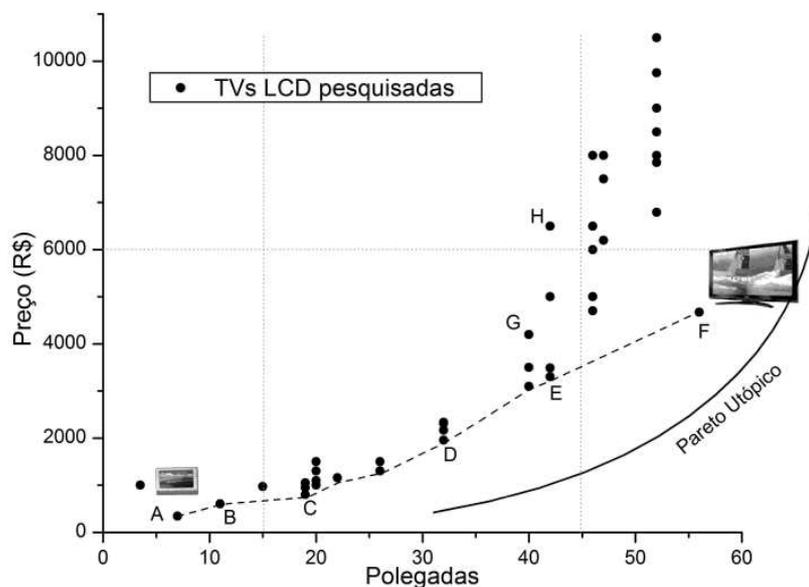


Figura 2.3. Gráfico formado entre o tamanho da tela e o preço de TVs LCD pesquisadas.

- Dominância de Pareto – se compararmos as soluções D e E, não se pode afirmar que uma é melhor que a outra (D e E são incomparáveis), pois enquanto D é mais barata, a solução E tem uma tela maior. Ao se comparar E e H, pode-se afirmar que E é melhor que H (E domina fracamente H), pois apesar de serem do mesmo tamanho, E consegue ter um melhor preço. Por outro lado, ao compararmos as soluções E e G, podemos

afirmar que E é muito melhor que G, pois além de ser um modelo mais em conta ainda possui uma tela maior (E domina G).

Assim, para duas soluções a e b (*i.e.* vetores de decisão) quaisquer em um problema de maximização, para todo $j = 1, 2, \dots, M$, a dominância de Pareto é definida:

$$\begin{aligned}
 a \succ b \text{ (a domina b)} & \quad \text{se e somente se } f_j(a) > f_j(b) \\
 a \succ b \text{ (a domina fracamente b)} & \quad \text{se e somente se } f_j(a) \geq f_j(b) \\
 a \sim b \text{ (a é incomparável com b)} & \quad \text{se e somente se } f_j(a) \geq f_j(b) \wedge f_j(b) \geq f_j(a)
 \end{aligned} \tag{2}$$

Ou seja, uma solução $x^{(1)}$ é dita como dominante de uma solução $x^{(2)}$, se e somente se, ambas as condições abaixo forem verdadeiras:

- i. A solução $x^{(1)}$ não é pior que $x^{(2)}$ para todos os objetivos;
 - ii. A solução $x^{(1)}$ é melhor que $x^{(2)}$ em pelo menos um objetivo.
- Pareto *front* – a linha tracejada representa o Pareto ótimo P_1 . Sobre ele são encontradas todas as soluções ótimas do problema. As soluções presentes no Pareto *front* representam os conflitos (*i.e. trade-off*) entre os objetivos, nenhuma solução melhorará em algum objetivo sem, necessariamente, piorar em outro objetivo – quaisquer duas soluções de P_1 são não-dominadas entre si. Todas as outras soluções fora do Pareto *front* são piores que alguma solução presente no *front*, em outras palavras, qualquer solução não Pareto é dominada por pelo menos uma solução Pareto $\{a, b \mid a \in P_1 \wedge b \notin P_1, a \succ b\}$ [Pareto1986]. Observe que o fato de existir mais de uma solução ótima comprova que o problema abordado é, de fato, MO. Caso estivéssemos procurando apenas a TV mais barata (solução A), ou quiséssemos a com maior tela (solução F), ou ainda, se o preço e o tamanho não fossem conflitantes, não estaríamos resolvendo um MOOP, mas um SOOP;
 - Pareto Utópico – a linha contínua forma o Pareto utópico, sobre o qual estariam as “impossíveis melhores soluções”. Elas representam os melhores resultados para todos os objetivos, ou seja, seria termos uma TV de 60” custando R\$100,00;
 - Pareto Real – dependendo do problema, em geral matemáticos, pode-se determinar precisamente como o Pareto é formado. A este Pareto verdadeiro, dá-se o nome de Pareto real;
 - Espaço de busca – as linhas pontilhadas delimitando R\$6000, 15” e 45” indicam os limites das variáveis, e dessa forma, criam as regiões viáveis e inviáveis delimitando o espaço de buscas;

- Conjunto de Aproximação – é o conjunto formado pelas soluções não-dominadas encontradas ao final de um processo de otimização por um algoritmo multi-objetivo, *i.e.* é o Pareto *front* construído por ele.

2.2.3 Busca e Tomada de Decisão

Dois tipos de problemas difíceis e conceitualmente distintos podem ser identificados quando se resolve um MOOP: (i) busca e (ii) tomada de decisão. O primeiro refere-se ao processo de otimização no qual o espaço de busca é explorado para encontrar soluções do Pareto *front*. Assim como em SOO, espaços de busca grandes e complexos podem dificultar as buscas e inviabilizar o uso de métodos exatos como programação linear [Steuer1986]. O segundo, relaciona-se com a seleção da solução mais adequada para a situação, solução essa que considere todos os objetivos, dentre o conjunto do Pareto *front*. Para avaliar os freqüentes e difíceis *trade-offs* entre objetivos conflitantes se faz necessário um tomador de decisão (do inglês, *decision maker*, DM), geralmente humano.

No contexto multi-objetivo pode-se perceber que o próprio espaço de busca é dividido em duas regiões: uma ótima e uma não-ótima. Existem, portanto, duas metas principais em uma MOO: (i) encontrar um conjunto de soluções o mais próximo possível do Pareto *front*; (ii) encontrar um conjunto de soluções o mais diverso (representativo entre os objetivos) possível. Estas duas metas, referem-se aos dois tipos de problemas comentados anteriormente: busca e tomada de decisão, respectivamente [Deb2001].

Dependendo de como a otimização e o processo de decisão são combinados, os métodos de MOO são classificados em três categorias [Hwang1979] [Horn1997]:

- a) Tomada de decisão antes da busca – os objetivos do MOOP são agregados em um único objetivo que implicitamente inclui as informações preferenciais do DM (ver Figura 2.4);

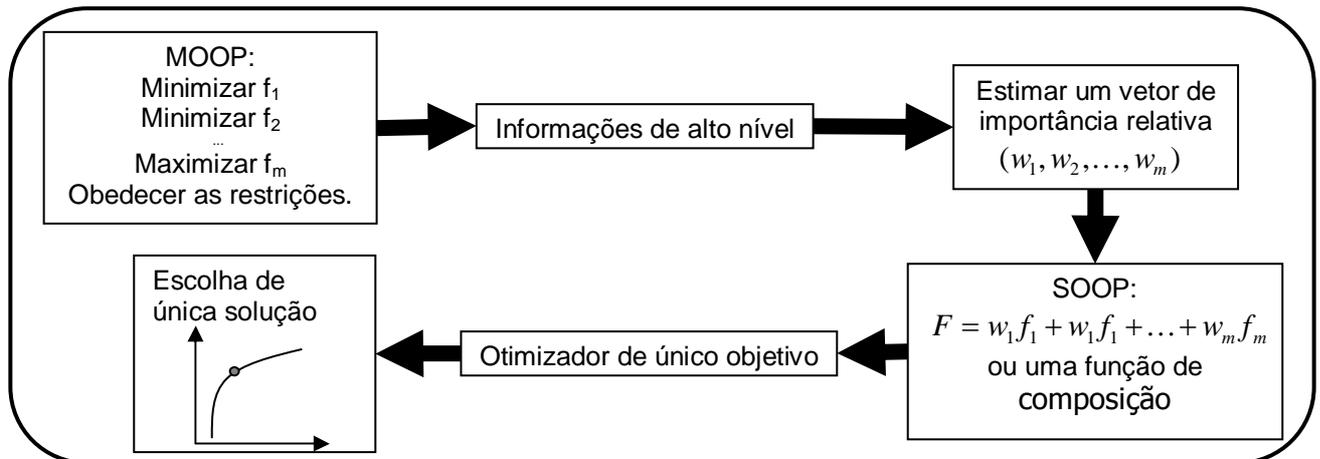


Figura 2.4. Esquema de decisão via um procedimento de otimização multi-objetivo baseado em preferências (adaptado de [Deb2001]).

- b) Busca antes da tomada de decisão – a otimização é realizada sem qualquer interferência, sem nenhuma informação preferencial. O resultado do processo de busca é um conjunto, idealmente o Pareto ótimo, de soluções candidatas entre as quais o DM toma sua decisão final (ver Figura 2.5);

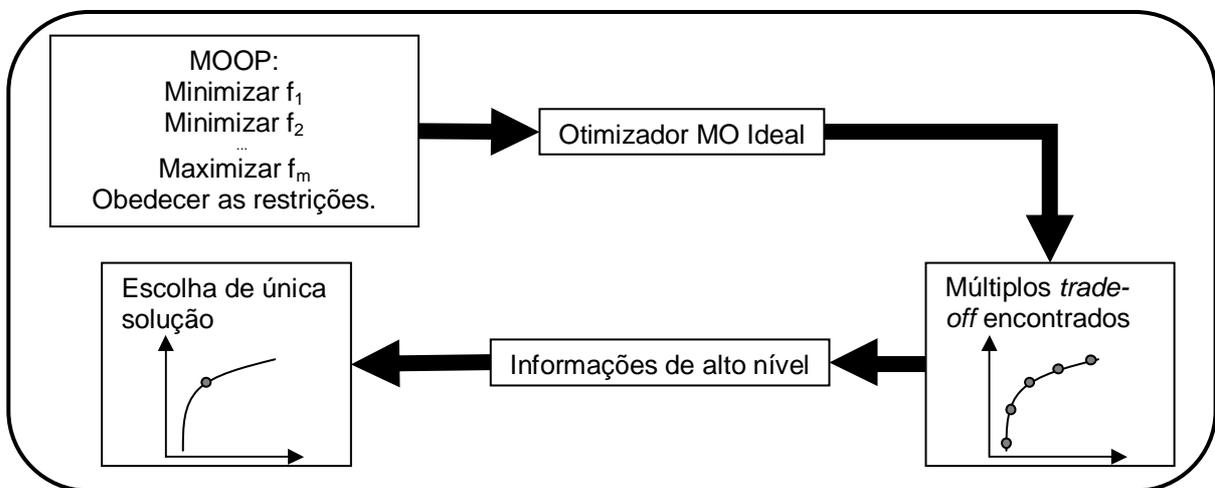


Figura 2.5. Esquema de decisão via um procedimento ideal de otimização multi-objetivo (adaptado de [Deb2001]).

- c) Tomada de decisão durante a busca – O DM pode articular suas preferências durante um processo de otimização interativo, comumente evolucionário. Após cada passo de otimização, o DM é apresentado a diversos *trade-offs* alternativos nos quais ele determina sua preferência; guiando assim a busca (ver Figura 2.6).

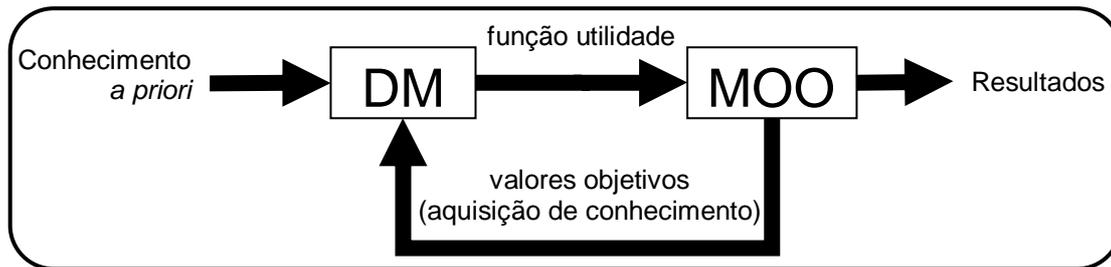


Figura 2.6. Esquema de decisão via um procedimento iterativo de otimização multi-objetivo (adaptado de [Fonseca1998a]).

A junção de múltiplos objetivos em um único critério de otimização oferece a vantagem de permutar a aplicação às estratégias clássicas de SOO sem maiores modificações. Entretanto, isso requer um profundo conhecimento do domínio de aplicação o que nem sempre ocorre. Realizar a busca antes da tomada de decisão evita esta dificuldade, mas exclui a articulação de preferências pelo DM o que poderia reduzir consideravelmente o espaço de busca. Uma outra dificuldade com esta e também com a terceira categoria de MOO pode ser a visualização e a apresentação dos conjuntos de soluções não-dominadas para MOOP de dimensões mais altas [Cohon1985]. Finalmente, a integração das buscas com a tomada de decisão é um caminho promissor para combinar as duas abordagens, unindo o melhor das duas [Zitzler1999a].

2.2.4 Por Que Otimização Evolucionária?

Os métodos clássicos resolvem otimizações multi-objetivo com abordagens baseadas em preferência, como visto na Figura 2.4. Assim, pode-se transformar um MOOP em um SOOP. Mas essa abordagem é ponto-a-ponto, ou seja, uma solução é modificada para criação de outra a cada iteração. Ao final do processo, tem-se uma única solução otimizada. Desse modo, para obtenção de um Pareto *front* são necessárias várias simulações, cada uma com um ajuste preferencial diferente. De acordo com Deb [Deb2001], a criação dessa abordagem preferencial para resolução de MOOP foi motivada pelo simples fato de limitação técnica. Na época, os algoritmos otimizadores eram construídos (e só conseguiam) para encontrar uma única solução por simulação.

Entretanto, a área de busca e otimização mudou bastante nos últimos anos em grande parte pela introdução de vários algoritmos de otimização não-clássicos, não-ortodoxos e estocásticos. Destes, os algoritmos evolucionários (do inglês, *evolutionary algorithms*, EA) são inspirados pelos princípios evolutivos da seleção natural de Darwin que guiam a busca por uma solução ótima. A maior diferença para os algoritmos clássicos é que os algoritmos evolucionários utilizam uma população de soluções ao invés de apenas uma única solução em cada iteração.

Assim, o resultado alcançado pelos EA também é uma população de soluções. Se um problema tem apenas um ponto ótimo (SOOP), espera-se que todos os indivíduos da população convirjam no mesmo ponto. Por outro lado, dado um MOOP, pode-se utilizar um EA para revelar múltiplas soluções ótimas em sua população final. Essa é a principal habilidade dos EA, e que os torna tão eficientes na resolução de MOOP.

Na Figura 2.7 estão indicados os vários tipos de algoritmos de otimização de acordo com uma classificação adaptada de Enrique Alba [Alba2007]. As buscas heurísticas surgiram com a necessidade de resolver problemas difíceis, *i.e.* de espaço de busca não-triviais, já que os algoritmos exatos não eram eficazes nesses domínios. Estas heurísticas são menos vulneráveis a forma e a continuidade do Pareto *front* (*e.g.* elas podem tratar facilmente Paretos descontínuos ou côncavos), o que é, de fato, um grande problema para as técnicas de programação matemática. Em adição, várias destas heurísticas são baseadas em população [Coello2007b].

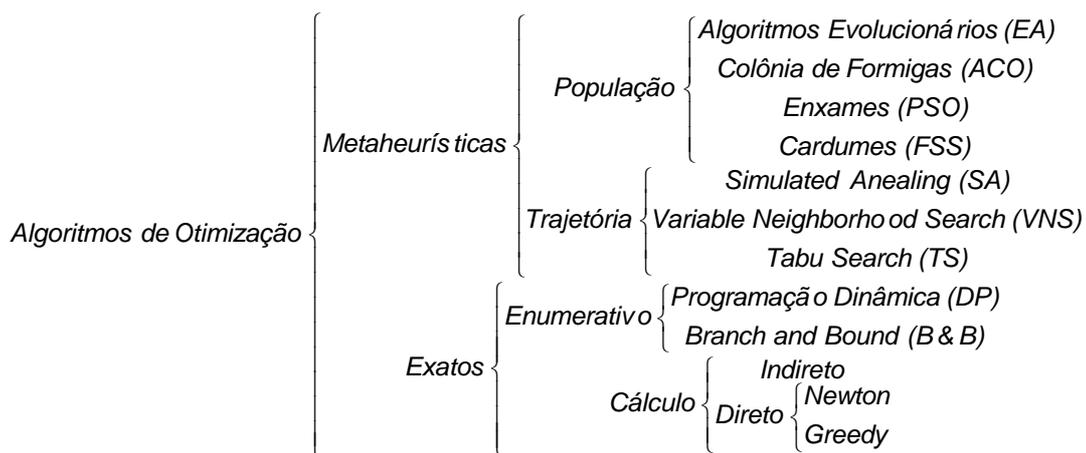


Figura 2.7. Diagrama de classificação de algoritmos de otimização, adaptado de [Alba2007].

Além dos EA, existem ainda outras heurísticas alternativas para busca e otimização, as quais são aplicadas nas mais diversas áreas: *simulated annealing* em redes de telecomunicações [Schuler2007], *tabu search* em tecnologias elétricas [Lin2008], colônia de formigas (do inglês, *ant colony optimization*, ACO) em *data-mining* [Wei2008], enxames (do inglês, *particle swarm optimization*, PSO) em problemas complexos de otimização [Carvalho2008], cardumes de peixes (do inglês, *fish school search*, FSS) em matemática [BastosFilho2008], sistemas imunológicos artificiais (SIA) em decisões executivas [Caldas2007], evolução diferencial em comunicações [Gupta2007], algoritmos culturais em arqueologia [Reynolds2008], entre outros.

2.2.5 A Ascensão dos Algoritmos Evolucionários Multi-Objetivo

Otimização Multi-Objetivo (sigla do inglês, MOO) não é um assunto novo na comunidade científica. As origens dos conceitos matemáticos para MOO podem ser datadas entre 1895 e 1906, com os trabalhos de Cantor e Hausdorff [O'Connor2007]. Apenas com a introdução do problema do vetor máximo (do inglês, *vector maximum problem*) de Kuhn & Tucker (1951) [Kuhn1951], MOO passou a ser considerada uma disciplina matemática em sua essência. A primeira aplicação conhecida na literatura foi um artigo de Zadeh no início da década de 60 [Zadeh1963]. Por outro lado, o uso de MOO tornou-se generalizado apenas na década de 70. Atualmente, existem mais de 30 técnicas matemáticas para MOO as quais referenciamos como algoritmos clássicos. A maioria destes algoritmos procura gerar elementos do Pareto ótimo individualmente. Entretanto, algumas destas técnicas podem não funcionar dependendo da forma do Pareto *front* [Coello2007b]. As primeiras aplicações em MOOP foram abordagens essencialmente baseadas em preferências, embora a necessidade de encontrar múltiplas soluções conflitantes já fosse claramente evidenciada.

A primeira aplicação real de EA que encontrava múltiplas soluções ótimas em apenas uma simulação foi sugerida e implementada por David Schaffer em sua dissertação de doutorado [Schaffer1984]. Sua criação, o *vector-evaluated genetic algorithm* (VEGA) foi uma simples modificação em um algoritmo genético (do inglês GA) de um único objetivo, mas demonstrou que GAs podiam ser utilizados para capturar várias soluções conflitantes em poucas iterações. Entretanto, se a otimização continuasse por um grande número de iterações, a população do VEGA tendia a convergir em uma solução única. Depois deste estudo inicial, passaram-se mais cinco anos até que surgisse algo realmente novo. Em 1989, David. E. Goldberg propôs o pseudocódigo de um plausível algoritmo evolucionário multi-objetivo (do inglês, *multi-objective evolutionary algorithm*, MOEA) utilizando o conceito de dominância [Goldberg1989]. Inspirados nesta publicação, diversos outros pesquisadores desenvolveram diferentes implementações de MOEA. Muitos destes novos MOEA foram imediatamente testados em diferentes problemas do mundo real para demonstrar que MOEA baseados na dominância de Pareto podem ser usados de maneira eficiente para encontrar e manter múltiplas soluções conflitantes [Ishibuchi1996] [Cunha1997] [Valenzuela1997] [Fonseca1998b] [Parks1998]; dentre estes estão: o GA multi-objetivo de Fonseca e Fleming (MOGA) [Fonseca1995]; o GA *non-dominated sorting* (NSGA) de Srinivas e Deb [Srinivas1994]; e o GA *niched* Pareto (NPGA) de Horn, Nafploitis e Goldberg [Horn1994]. Nesta mesma época, outros pesquisadores propuseram diferentes maneiras de resolver MOOP usando um EA. Alguns exemplos são: a abordagem diplóide de Kursawe

[Kursawe1990]; o GA baseado em pesos de Hajela e Lin [Hajela1992]; e o GA baseado em distância de Osyczka e Kundu [Osyczka1995]. Nos anos seguintes, alguns pesquisadores investigaram aspectos particulares de MOOP, tais como: convergência para o Pareto *front* ótimo [Veldhuizen1998] [Rudolph1998]; *niching (fitness sharing)* [Obayashi1998]; e elitismo [Parks1998] [Obayashi1998]; enquanto outros concentraram-se em desenvolver outras técnicas [Lis1997] [Laumanns1998]. Ainda nesse período, diversos artigos de revisão e tutoriais foram disponibilizados [Fonseca1995] [Tamaki1996] [Horn1997] [Veldhuizen1998] [Deb1999] [Coello1999]. Com o aumento de interesse em MOO, revistas científicas indexadas sobre EA começaram a incluir debates especiais [Deb2000b], conferências de EA começaram a contemplar tutoriais e seções especiais em otimização evolucionária multi-objetivo (do inglês, *evolutionary multi-objective optimization*, EMOO), e em 2001 iniciou-se uma conferência internacional independente em EMOO [Zitzler2001a]. O que se viu na seqüência foi um real amadurecimento de EMOO, a partir do grande volume de pesquisas, problemas mais complexos foram solucionados.

Em geral, existem dois tipos de MOEA: os que não incorporam o conceito de dominância de Pareto nos mecanismos de seleção (*i.e.* abordagens que usam funções lineares agregadoras) e os que classificam a população baseados no Pareto. No entanto, historicamente esta separação pode ser feita em duas gerações:

- Primeira Geração - algoritmos relativamente simples, abordagens baseadas em Pareto, baseadas em população e outras mais simples que usam funções agregadoras. Os algoritmos mais representativos são [Deb2001]: VEGA, MOGA, NSGA e NPGA;
- Segunda Geração – algoritmos incorporam o conceito de elitismo de duas formas: preservando melhores soluções na seleção ou usando uma segunda população. Representando esta geração temos [Deb2001]: SPEA e SPEA2, NSGA-II, PAES, PESA, PESA-II, μ GA2, entre outros.

Após dez anos de sucesso os algoritmos da primeira geração tornaram-se obsoletos e os de segunda geração são agora considerados o estado da arte em EMOO. Esta segunda geração enfatizou a eficiência computacional e agora, problemas como dimensionalidade são considerados. A Figura 2.8 mostra o crescente número de pesquisas e publicações na área (até o início de 2007).

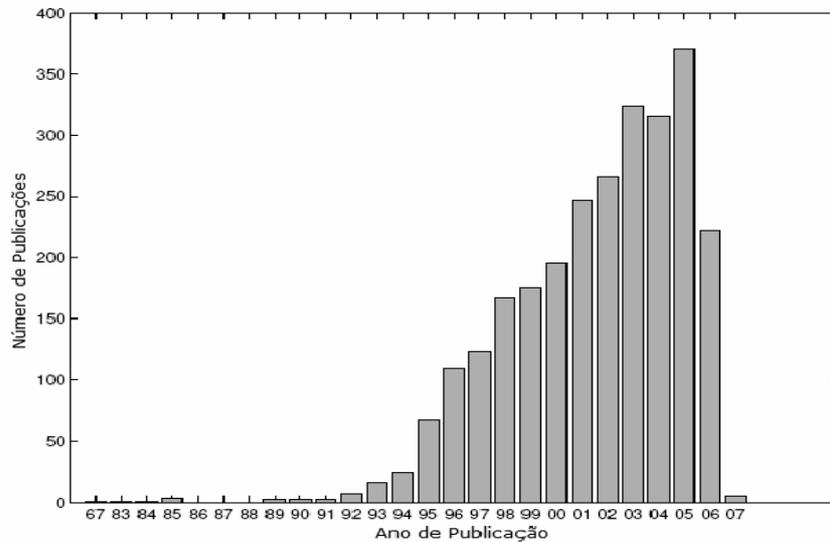


Figura 2.8. Crescente número de publicações de estudos em MOEA de 1967 até o início de 2007 (adaptado de [Coello2007b]).

Os algoritmos de segunda geração foram propostos, basicamente, com dois grandes objetivos: aproximar-se do Pareto real e manter uma boa diversidade por todo este Pareto [Deb2001]. Por outro lado, estudos demonstraram que quando tratam de problemas com muitos objetivos - a partir de cinco, os MOEAs não conseguem manter essa diversidade além de apresentarem um alto custo computacional [Fleming2005].

As técnicas de computação evolucionária não foram concebidas para lidar com restrições [Fonseca1998a] de modo que foram criados vários artifícios para essa inclusão, dos quais destacamos: funções de penalidade, decodificadores, heurísticas e incorporação de preferências. Coello enfatizou a importância e o constante desenvolvimento nos últimos anos de algoritmos para o tratamento de restrições [Coello2000] [Coello2007a], mas a abordagem de Deb [Deb2002] ainda é uma das mais divulgadas. Esta abordagem é uma simplificação da proposta de Fonseca e Fleming [Fonseca1998a], que trata as restrições como objetivos mais prioritários (*i.e. hard objectives*) além de ser capaz de incorporar prioridades, metas e preferências. No tratamento de problemas reais com muitos objetivos, onde o decisor pode interagir com preferências, a abordagem de Fonseca e Fleming, que é mais completa, continua sendo aplicada [Lygoe2005] [Fonseca2007].

2.2.6 Principais Algoritmos Multi-Objetivo Usados Atualmente

Como o foco deste trabalho inclui os MOEA, estes serão mais detalhados que os algoritmos clássicos para tratar MOOP.

A. Algoritmos Clássicos

Existem, basicamente três categorias de métodos de resolução de MOOP. A primeira delas é uma abordagem elementar, que a depender do problema, pode gerar resultados satisfatórios. Os principais métodos desta categoria são: análise de prós-e-contras (*Pros-Cons Analysis*); métodos *Maximin & Maximax*; métodos conjuntivos e disjuntivos (*conjunctive and disjunctive*); e o método Lexicográfico (*Lexicographic*) [Linkov2004].

A segunda categoria é baseada na Teoria da Utilidade de Multi-Atributos (do inglês, *Multi-Attribute Utility Theory*, MAUT) e tem como meta agregar diferentes objetivos em uma função única, a qual deverá ser maximizada ou minimizada. As principais técnicas deste grupo são: *MAUT Additive Linear*, *SMART*, *Generalized Means* e o método de análise hierárquica (do inglês, *Analytic Hierarchy Process*, AHP) [Fülöp2005].

A última categoria inclui os métodos *outranking*. Estes seguem o conceito de mesmo nome proposto por Roy (1968). A idéia básica é criar subconjuntos tão pequenos quanto possíveis de acordo com essa classificação (*outranking*). Os métodos que representam esta categoria são subdivididos em duas famílias: *ELECTRE* e *PROMETHEE* [Figueira2004].

B. Algoritmos Evolucionários

I) *Vector Evaluated Genetic Algorithm (VEGA)*

Foi proposto por Schaffer [Schaffer1984] [Schaffer1985]. Apenas o mecanismo de seleção do GA é modificado, de modo que a cada geração várias subpopulações são geradas a partir de uma seleção proporcional de acordo com cada função objetivo em questão. Estas subpopulações são misturadas para formar a nova população, sobre a qual os operadores de cruzamento e mutação serão aplicados normalmente. Suas vantagens são a eficiência e a facilidade de implementação, por outro lado, o mecanismo de seleção desenvolvido, em certas condições, se comporta da mesma maneira que as abordagens com função de agregação e, conseqüentemente, está sujeito às mesmas limitações destas [Coello2007b].

II) *Nondominated Sorting Genetic Algorithm (NSGA)*

Foi proposto por Srinivas e Deb [Srinivas1994]. É baseado na classificação dos indivíduos em várias camadas. É atribuído um *fitness* específico para os indivíduos não-dominados e depois estes são removidos da população. O processo é repetido até que toda a população seja classificada. Para manter a diversidade na população, os indivíduos classificados são agregados no espaço de decisão de acordo com o *fitness* atribuído. Apesar da relativa

facilidade de implementação, parece ser muito sensível ao fator de agregação (*i.e. sharing factor*) [Coello2007b].

III) Strength Pareto Evolutionary Algorithm (SPEA)

Foi introduzido por Zitzler e Thiele [Zitzler2001a]. Usa um arquivo externo que guarda as soluções não-dominadas encontradas durante o processo evolucionário. Calcula o valor da força (*strength*) dos indivíduos de maneira parecida a classificação realizada pelo algoritmo MOGA [Fonseca1995], ou seja, o rank de classificação de determinado indivíduo corresponde ao número de indivíduos dominados por ele na população corrente. A técnica de clustering *average linkage method* (Morse, 1980) é usada para manter a diversidade.

IV) Strength Pareto Evolutionary Algorithm 2 (SPEA2)

Uma versão modificada do SPEA com três principais diferenças: (i) uma atribuição de *fitness* mais refinada que para cada indivíduo considera o número de dominados e o número de dominadores; (ii) utiliza a técnica de estimativa de densidade do vizinho mais próximo (*nearest neighbor*) guiando a busca mais eficientemente; e (iii) possui um melhor método de truncamento de arquivo, o que garante a preservação das soluções limites [Zitzler2001b]. A Figura 2.9 ilustra o pseudo-código do algoritmo SPEA2.

```
1 // Entradas: P[0] (população inicial), A (arquivo externo) e tmax (número máximo
2 // de gerações).
3 t = 0;
4 While (t < tmax){
5     // Atribuí fitness para todos os indivíduos da união da população atual + o
6     // arquivo externo.
7     (P[t] + A).calculateFitness();
8     // Utiliza operadores de seleção, cruzamento e mutação para criar a próxima
9     // geração.
10    P[t+1] = (P[t] + A).selection().crossover().mutation();
11    // Atualiza o arquivo externo com as melhores soluções (não-dominadas) e
12    // removendo as soluções dominadas.
13    A.update();
14    t++;
15 } //End while
16 // Retorna o arquivo externo com as melhores soluções.
17 return A;
```

Figura 2.9. Pseudo-código do algoritmo SPEA2.

V) Nondominated Sorting Genetic Algorithm II (NSGA-II)

Deb *et al.* propuseram uma nova versão do NSGA que é mais eficiente e menos complexa, computacionalmente falando. Incorpora elitismo e o operador *crowding distance* que

consegue manter a diversidade das soluções sem a especificação de qualquer parâmetro adicional [Deb2000a] [Deb2002]. A Figura 2.10 exhibe o pseudo-código do algoritmo NSGA-II.

```

1 // Entradas:P[0],Q[0] populações iniciais, N (tamanho da população), tmax (número
2 // máximo de gerações).
3
4 t=0
5 While (t < tmax){
6     // Criação de Rt com a união das populações pai e filha.
7     Rt = P[t] + Q[t];
8     // Realiza o ordenamento não-dominado (non-dominated sorting) em Rt,
9     // identificando os diferentes níveis de fronts: Fi, i = 1, 2, ...
10    // utilizando o algoritmo de ranking.
11    Array F[] = Rt.sorting();
12    P[t+1] = P[t];
13    i = 1;
14    While (P[t+1].size() + F[i].size) < N){
15        P[t+1] += F[i]
16        i++;
17    }//End while
18    // Inclui na população filha as (N - P[t+1].size()) soluções mais espalhadas
19    // utilizando o operador de crowding.
20    F[i].crowding();
21    P[t+1] += F[i];
22    // Criação da próxima geração Q[t+1] a partir de P[t+1] utilizando operadores
23    // de seleção (torneio binário), cruzamento e mutação.
24    Q[t+1] = P[t+1].selection().crossover().mutation();
25    t++;
26 } //End while
27 // Retorna o primeiro front da união da população filha e pai da ultima geração.
28 F[] = (P[tmax] + Q[tmax]).sorting();
29 return F[1];

```

Figura 2.10. Pseudo-código do algoritmo NSGA-II.

VI) Pareto Envelop-based Selection Algorithm (PESA)

PESA foi proposto por Corne *et al.* [Corne2000]. Essa abordagem utiliza uma pequena população interna e uma população externa (ou secundária) maior. PESA usa a mesma hipergrade (*hyper-grid*) de divisão no espaço objetivo para manter a diversidade que a adotada pelo algoritmo PAES [Knowles1999]. Ele consiste de um procedimento para manter a diversidade que divide o espaço objetivo de maneira recursiva, onde cada solução é posicionada em determinado *grid* de acordo com os valores dos seus objetivos. Um mapeamento deste *grid* é mantido indicando a quantidade de soluções existentes dentro de cada um. Esta metodologia é utilizada para decidir que soluções serão introduzidas na população externa (*i.e.* soluções não-dominadas encontradas ao longo do processo). Este procedimento é adaptativo e nenhum parâmetro adicional é necessário (apenas o número de divisões no espaço objetivo).

VII) Pareto Envelop-based Selection Algorithm-II (PESA-II)

É uma revisão do PESA na qual uma seleção baseada em região é adotada. Nesta seleção, a unidade básica de seleção é um hipercubo⁴ (*hyperbox*) ao invés de um indivíduo. O procedimento consiste em selecionar um hipercubo utilizando qualquer técnica tradicional de seleção. Em seguida, caso existam mais de um indivíduo neste hipercubo, seleciona-se um deles aleatoriamente [Corne2001]. A Figura 2.11 exibe o pseudo-código deste algoritmo.

```
1 // Entradas: p (tamanho da população)
2 X* = {};
3 Xpop = createPop(p);
4 // enquanto a condição de parada não estiver satisfeita
5 while (!terminationCriterion()){
6     for( int i = 0; i < Xpop.size(); i++){
7         // calcula o fitness.
8         X* = updateOptimalSet(X*, Xpop[i]);
9     }
10    // remove soluções dominadas.
11    X* = pruneOptimalSet(X*);
12    // realiza a seleção através de hipercubos.
13    // cria a próxima geração a partir de cruzamento e mutação
14    Xpop = pesaSelection(X*, p).crossover().mutation();
15 } // end while
16 // O conjunto com os melhores indivíduos encontrados.
17 return X*;
```

Figura 2.11. Pseudo-código do algoritmo PESA-II.

2.2.7 Métricas de Desempenho

Os primeiros MOEA demonstraram seus desempenhos a partir de ilustrações no espaço objetivo das soluções não-dominadas encontradas e registradas juntamente com o Pareto ótimo real. A ênfase nesses estudos era mostrar como a otimização havia se aproximado do Pareto real. Posteriormente surgiu a necessidade de se quantificar melhor os desempenhos em diversos problemas de teste.

Reiterando o afirmado anteriormente, uma MOO possui duas metas ortogonais: (i) encontrar soluções o mais próximo do Pareto *front* possível; e (ii) encontrar soluções o mais diversificado possível sobre o Pareto (ver Figura 2.12). Por sua vez, a métrica diversidade pode ser classificada em duas: extensão (*i.e.* no sentido de espalhamento das soluções extremas) e distribuição (*i.e.* no sentido de distância relativa entre as soluções) [Zitzler2000].

⁴ Hipercubo é uma proposição teórica de um poliedro de quarta dimensão.

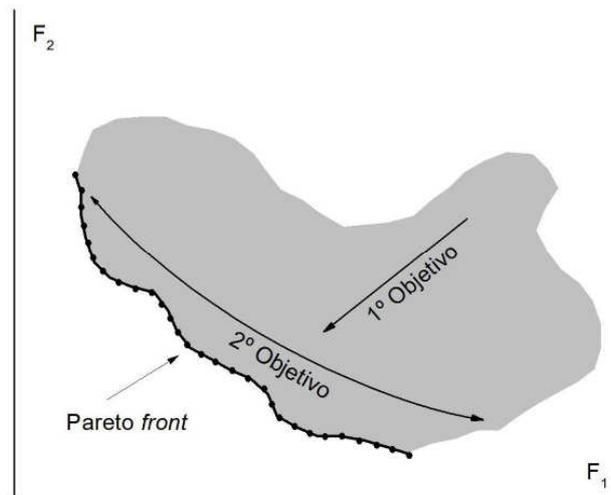


Figura 2.12. Dois objetivos de MOO e um conjunto de soluções não-dominadas ideal em um problema hipotético, adaptado de [Deb2001].

Um MOEA será classificado como bom se satisfizer ambos os objetivos, *i.e.* se conseguir se aproximar do Pareto *front* com uma boa diversidade de soluções; a Figura 2.12 traz também um exemplo de um conjunto de soluções não-dominadas para um problema hipotético. No entanto, essa configuração ideal nem sempre pode ser alcançada; em determinados problemas sequer conhecemos o Pareto *front* real. Quando comparamos dois conjuntos de aproximação outros cenários podem ser observados, por exemplo, parte de um conjunto domina parte do outro e vice-versa. Essas situações podem criar uma terceira variante na análise de MOEA. Desse modo, para que as comparações possam ser realizadas, os conjuntos de aproximação devem ser transformados em números através das métricas. Ainda assim, a análise de um MOEA é uma tarefa multi-objetivo e requer pelo menos duas métricas diferentes. Existem métricas que medem (i) a proximidade com o Pareto, (ii) a diversidade das soluções e (iii) que medem ambas implicitamente. Descrevemos a seguir, em detalhes, cada uma das três categorias.

A. Métricas Que Avaliam a Proximidade ao Pareto

Estas métricas calculam explicitamente a proximidade de um conjunto Q de N soluções a um conhecido Pareto *front* P^* . O conjunto P^* pode conter infinitas soluções - quando se conhece a equação que descreve o relacionamento entre as variáveis de decisão, ou não - quando apenas algumas soluções são conhecidas ou possíveis de computar. As métricas mais usadas nesta categoria são: *error ratio* [Veldhuizen1999], *set coverage* [Zitzler1999a], *generational distance* [Veldhuizen1999] e *maximum Pareto-optimal front error* [Veldhuizen1999]. Detalharemos a métrica *Set Coverage*. Está métrica também é conhecida como métrica C e é similar a métrica proposta por Zitzler [Zitzler1999a]. Também pode ser usada para se ter uma

idéia do espalhamento relativo entre dois conjuntos de aproximação A e B . A métrica $C(A, B)$ calcula a proporção de soluções em B que são fracamente dominadas pelas soluções de A .

$$C(A, B) = \frac{|\{b \in B \mid \exists a \in A \therefore a \succ b\}|}{|B|}. \quad (3)$$

Se a métrica assume o valor $C(A, B)=1$ significa que todos os membros de B são fracamente dominados por A . Por outro lado, $C(A, B)=0$ significa que nenhum membro de B é fracamente dominado por A . Como o operador de dominância não é simétrico, $C(A, B)$ não necessariamente é igual a $1 - C(B, A)$; deve-se, portanto, calcular ambas as formas para entender quantas soluções de A são cobertas por B e vice-versa [Deb2001]. Note que a cardinalidade dos dois conjuntos não precisa ser igual.

Embora Zitzler use essa métrica para comparar o desempenho de dois algoritmos, ela também pode ser utilizada para medir o desempenho de um algoritmo, fazendo $A = P^*$ e $B = Q$. A métrica $C(P^*, Q)$ irá determinar a proporção de soluções em Q que são fracamente dominadas pelos membros do Pareto *front* P^* . É desnecessário escrever que $C(Q, P^*)$ será sempre igual a zero.

B. Métricas Que Avaliam a Diversidade ao Longo do Pareto

Também existem métricas que calculam apenas a diversidade entre as soluções não-dominadas do conjunto de aproximação obtido pelo MOEA. As mais importantes são: espaçamento (*spacing*) [Schott1995], espalhamento (*spread*) [Deb2000a], espalhamento generalizado (*generalized spread*) [Zhou2006], máximo espalhamento (*maximum spread*) [Zitzler1999a] e *chi-square-like deviation measure* [Deb1989]. Em especial vamos descrever a métrica *Spread*. Deb *et al.* [Deb2000a] propuseram esta métrica para atenuar as dificuldades presentes na métrica *spacing* [Schott1995].

$$\Delta = \frac{\sum_{m=1}^M d_m^e + \sum_{i=1}^{|Q|} |d_i - \bar{d}|}{\sum_{m=1}^M d_m^e + |Q| \bar{d}}, \quad (4)$$

onde d_i pode ser qualquer medida de distância entre soluções vizinhas e \bar{d} é a média dessas distâncias. A distância Euclidiana, a soma da diferença absoluta entre os objetivos ou o operador de *crowding distance* podem ser usados para calcular d_i . O parâmetro d_m^e representa a distância entre as soluções mais extremas de P^* e Q correspondente a m -ésima função objetivo. Para uma distribuição ideal de soluções, $\Delta = 0$.

C. Métricas Que Avaliam a Proximidade e a Diversidade Simultaneamente

Existem ainda algumas métricas onde ambos os objetivos da MOO são avaliados de maneira combinada. Estas métricas podem fornecer medidas qualitativas para a convergência assim como para a diversidade. Entretanto, quando combinadas com qualquer uma das métricas citadas anteriormente, obtém-se uma melhor avaliação geral. As representantes desta categoria são: hipervolume [Veldhuizen1999] [Zitzler1998], superfície alcançada baseada em estatística (do inglês, *attainment surface based statistical metric*) [Fonseca1996], métrica ponderada (do inglês, *weighted metric*) [Deb2001], métrica de avaliação não-dominada (do inglês, *non-dominated evaluation metric*) [Deb2001], entre outras [Hansen1998]. Aqui, o destaque será dado ao Hipervolume (HV), que será amplamente utilizado na contribuição deste trabalho.

Esta métrica calcula o volume do espaço objetivo coberto (explorado) pelas soluções de Q para problemas onde todos os objetivos são minimizações [Veldhuizen1999] [Zitzler1998]. Matematicamente, para cada solução $i \in Q$, um hipercubo v_i é construído em torno da diagonal formada entre um ponto de referência W e a solução i . Este ponto de referência (vetor de objetivos) pode ser simplesmente construído a partir dos piores valores para cada uma das funções objetivo. Depois, une-se todos os hipercubos encontrados e calcula-se o hipervolume:

$$HV = \text{volume}\left(\bigcup_{i=1}^{|Q|} v_i\right). \tag{5}$$

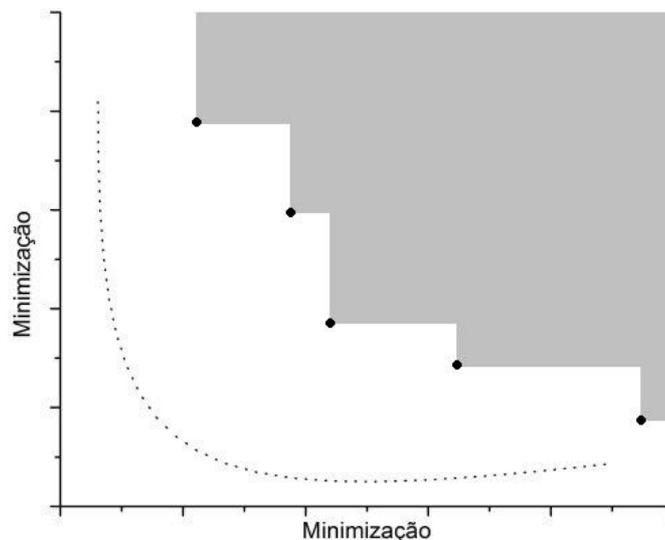


Figura 2.13. Um exemplo da representação da métrica hipervolume. Em cinza, a região do espaço de busca explorada pela população.

Obviamente que maiores valores de HV são desejados. No entanto, essa métrica é vulnerável a escala dos objetivos. A Figura 2.13 ilustra como o hipervolume analisa a área do espaço de busca que foi explorada em um processo de otimização. Por exemplo, no problema

hipotético da Figura 2.12, se o primeiro objetivo assume valores muito maiores que o segundo, uma unidade de melhora em F_1 melhoraria muito mais o HV que uma unidade de melhora em F_2 . Assim, essa métrica favorecerá o conjunto Q que tiver uma melhor convergência para o objetivo com a maior escala. Para suprimir esse problema, a métrica deve ser aplicada com os valores dos objetivos normalizados.

Outra maneira de eliminar esse problema de escalas diferentes e calcular um HV normalizado é usar a métrica HVR que é a taxa (do inglês, *ratio*) do HV de Q em P^* [Veldhuizen1999]:

$$HVR = \frac{HV(Q)}{HV(P^*)}. \quad (6)$$

Para problemas com minimizações em todos os objetivos, o melhor valor para a métrica é $HVR = 1$ (quando $Q = P^*$). Assim, quanto mais perto de 1 for o HVR , mais próximo do Pareto *front* estará o conjunto de aproximação obtido.

2.3 Problemas da Mochila

Nesta seção descreve-se o paradigma dos problemas da mochila. Este paradigma combinatorial é bastante utilizado na equivalência de problemas reais. Neste trabalho, em especial, o problema da mochila é comparado com a colheita de cana-de-açúcar. Descrevem-se ainda como os algoritmos evolucionários têm contribuído nesta classe de problemas e como estes estudos podem ser reutilizados no escopo deste mestrado.

2.3.1 História

Problemas da Mochila (do inglês, *knapsack problems*, KP) têm sido estudados intensamente, desde o trabalho pioneiro de Dantzig na década de 50 [Dantzig1957], por dois grandes motivos: (i) sua aplicação imediata em problemas industriais e financeiros; e, principalmente, (ii) por razões teóricas, já que o KP são freqüentemente associados a diversos problemas de programação inteira [Schrijver1986].

Em toda família de problemas da mochila, pressupõe-se que um subconjunto de objetos (ou itens) seja escolhido de modo que a soma de seus valores seja máxima e que a capacidade da(s) mochila(s) não seja(m) ultrapassada(s). Dependendo da disposição dos objetos e das mochilas, diferentes tipos de problemas são instanciados: no Problema da Mochila 0/1 (*0/1 Knapsack Problem*, KP) cada item deve ser escolhido no máximo uma vez, enquanto que no

Problema da Mochila Limitado (*Bounded Knapsack Problem*, BKP) tem-se um limite de quantidade para cada item. O Problema da Mochila de Múltiplas Escolhas (*Multiple-Choice Knapsack Problem*, MChKP) ocorre quando os objetos devem ser escolhidos de classes disjuntas. Ainda, se existem várias mochilas que devem ser preenchidas simultaneamente, configura-se o Problema da Mochila Múltiplo (*Multiple Knapsack Problem*, MKP). A forma mais geral é o Problema da Mochila de Múltiplas Restrições (*Multiple-Constrained Knapsack Problem*, MCoKP), que é basicamente um Problema de Programação Inteira (do inglês, *integer programming problem*, IPP) genérico com coeficientes inteiros [Pisinger1995].

Apesar de suas simples definições, toda a família de problemas KP é NP-completo [Papadimitriou1982], ou seja, é pouco provável que se consiga desenvolver algoritmos que resolvam estes problemas em tempos polinomiais. Porém, mesmo com o tempo de execução exponencial no pior caso, muitas instâncias de alta escala conseguem ser solucionadas em frações de segundos. Estes resultados são frutos de várias décadas de pesquisa as quais revelaram propriedades estruturais dos KP que tornaram possíveis suas resoluções [Pisinger1995]. As principais técnicas utilizadas no desenvolvimento dessa área nas últimas décadas foram: *branch-and-bound* [Kolesar1967], programação dinâmica [Gilmore1963], *state space relaxation* [Ibaraki1987a] e *preprocessing* [Ibaraki1987b]. A partir do desenvolvimento de técnicas heurísticas, em especial com computação evolutiva, os MOEA passaram a ser bastante utilizados na solução dessa classe de problemas [Zitzler1999a].

2.3.2 Formulação

Em todas as variantes do problema existem alguns objetos com um *valor* p_j e um *peso* w_j que são guardados em uma ou mais mochilas de *capacidade* c . Assume-se que todos os coeficientes são inteiros positivos embora suposições mais simples, por vezes, são necessárias em problemas individuais.

O KP 0/1 é o problema de escolher um subconjunto de n objetos tal que a soma dos valores destes objetos seja maximizada e que a soma dos respectivos pesos não ultrapasse a capacidade c . Assim, o problema pode ser formulado com a seguinte maximização, onde x_j é uma variável binária igual a 1 se o item foi incluído na mochila e igual a 0 caso contrário. Se o valor p_j é igual ao peso w_j , tem-se o Problema *Subset-sum*.

$$\begin{aligned} & \text{maximize } \sum_{j=1}^n p_j x_j \\ & \text{restrito a } \sum_{j=1}^n w_j x_j \leq c \end{aligned} \quad x_j \in \{0,1\}, \quad j=1,\dots,n. \quad (7)$$

Se existe uma quantidade limite m_j por cada objeto j , configura-se o BKP. Aqui, x_j é a quantidade de cada objeto incluída na mochila de modo que se consiga o maior valor acumulado:

$$\begin{aligned} & \text{maximize } \sum_{j=1}^n p_j x_j \\ & \text{restrito a } \sum_{j=1}^n w_j x_j \leq c \end{aligned} \quad x_j \in \{0,1,\dots,m_j\}, \quad j=1,\dots,n. \quad (8)$$

O Problema da Mochila Ilimitado (*Unbounded Knapsack Problem*, UKP) é uma generalização deste limitado, onde existe uma quantidade ilimitada de objetos disponíveis. Em última instância, qualquer variável x_j de um problema ilimitado será ainda limitada pela capacidade da mochila. Assim, em geral não existe vantagens em transformar um problema ilimitado na versão limitada [Pisinger1995].

Imagine agora que um caixa precisa devolver trocos usando a menor quantidade possível de moedas w_1, \dots, w_n . Define-se o *Change-making Problem*, onde w_j é o valor da moeda j ; assume-se que existe uma quantidade ilimitada de moedas disponíveis. O número ótimo de cada moeda j usada é expresso por x_j . Esse problema é considerado uma minimização variante do UKP, onde $p_j = 1$ e a restrição é uma igualdade com a capacidade.

Uma outra generalização do KP é quando é preciso escolher exatamente um objeto j de cada uma das k classes N_i , $i=1,\dots,k$, tal que a soma dos valores seja maximizada. Isto configura o MChKP. Aqui a variável binária $x_{ij} = 1$ significa que o objeto j foi escolhido da classe i . A restrição $\sum_{j \in N_i} x_{ij} = 1$, $i=1,\dots,k$, garante que exatamente um objeto de cada classe será escolhido:

$$\begin{aligned} & \text{maximize } \sum_{i=1}^k \sum_{j \in N_i} p_{ij} x_{ij} \\ & \text{restrito a } \sum_{i=1}^k \sum_{j \in N_i} w_{ij} x_{ij} \leq c, \\ & \sum_{j \in N_i} x_{ij} = 1, \end{aligned} \quad \begin{aligned} & x_j \in \{0,1\}, \\ & i=1,\dots,k, \\ & j=1,\dots,n, \quad j \in N_i. \end{aligned} \quad (9)$$

Se escolhermos n objetos para serem empacotados em m mochilas com possíveis diferentes capacidades c_i , de modo que a maior soma possível de valores seja obtida, teremos o

MKP. Aqui $x_{ij} = 1$ indica que o objeto j foi empacotado na mochila i , enquanto a restrição $\sum_{j=1}^n w_j x_{ij} \leq c_i$ garante que a capacidade de todas as mochilas é respeitada. Já a restrição $\sum_{i=1}^m x_{ij} \leq 1$, garante que cada item será escolhido no máximo uma vez. Uma generalização desse problema será usada neste mestrado para resolver o problema da colheita da cana-de-açúcar.

$$\begin{aligned} & \text{maximize} \quad \sum_{i=1}^m \sum_{j=1}^n p_j x_{ij} \\ & \text{restrito a} \quad \sum_{j=1}^n w_j x_{ij} \leq c_i, \quad i = 1, \dots, m, \\ & \quad \quad \quad \sum_{i=1}^m x_{ij} \leq 1, \quad j = 1, \dots, n. \end{aligned} \quad (10)$$

A forma mais geral de um KP é o MCoKP que é uma forma genérica do IPP onde todos os coeficientes p_j , w_{ij} e c_i são inteiros não-negativos. Ele pode ser formulado da seguinte forma:

$$\begin{aligned} & \text{maximize} \quad \sum_{j=1}^n p_j x_j \quad x_j \geq 0, \\ & \text{restrito a} \quad \sum_{j=1}^n w_{ij} x_j \leq c_i \quad i = 1, \dots, m, \\ & \quad \quad \quad j = 1, \dots, n. \end{aligned} \quad (11)$$

Existem ainda outros problemas da família KP, entre eles citamos: o *Bin-packing Problem* [Pisinger1995], o *Collapsing KP* [Fayard1994], o *Nested KP* [Dudzinski1987], o *Nonlinear KP* [Morin1976] e o *Inverse-parametric KP* [Burkard1994].

2.3.3 O Problema da Mochila e os Algoritmos Evolucionários

As possibilidades de instanciação do KP em problemas reais eram evidentes, de modo que com o desenvolvimento dos EA, estes algoritmos estocásticos foram amplamente utilizados. Na maioria dos estudos, os pesquisadores utilizavam uma representação binária para modelar o KP em algoritmos genéticos. Devido à própria formulação matemática dos KP, é fácil mapear um indivíduo binário no qual os bits ativados representam os objetos escolhidos para a mochila.

Existem diversos artigos com aplicações que utilizam apenas essa representação binária na resolução dos problemas KP [Watannabe1992] [Olsen1993] [Gordon1993a] [Gordon1993b] [Khuri1994]. Os primeiros trabalhos que aplicaram EA em KP com representações diferentes foram os de Hinterding [Hinterding1994] e Michalewicz [Michalewicz1996]. Em seu trabalho seguinte Hinterding [Hinterding1999], formalizou diversas formas de representação de problema e de tratamento de restrições na utilização de EA em KP.

Baseada nesses trabalhos anteriores, Christine Mumford estendeu as representações desenvolvidas para o simples KP e as remodelou para o KP multi-objetivo, o MKP. Nesse estudo, além de comparar diferentes representações, ela também investigou como diferentes operadores de cruzamento influenciavam o problema [Mumford2003]. Em seguida, Colombo e Mumford incluíram diferentes MOEA [Colombo2005] nessas investigações.

Com isso, definiram-se três classes ou formas de representação para KP e três grupos de tratamento de restrição:

A. Tipos de Representação

- Representação binária – é aquela onde um bit é valorado para 1 indicando que o objeto daquela posição foi selecionado para a mochila. Dessa forma, os EA podem selecionar os objetos diretamente, por outro lado, podem precisar de mecanismos que corrijam ou validem as mochilas produzidas;
- Representação numérica – aqui os genes dos indivíduos dos GA são números e não bits. Para isso, necessita-se, primeiramente, decodificar o cromossomo e gerar uma ordem nos objetos. Na seqüência, utiliza-se algum procedimento para produzir mochilas válidas (*i.e.* que respeitam as restrições de capacidade). Dois métodos foram propostos: (i) representação ordinal [Michalewicz1996] e (ii) representação de parâmetro (*random key*) [Hinterding1999];
- Representação simbólica – onde os genes representam os próprios objetos. Também foram propostas duas abordagens [Hinterding1999]: (i) representação *order based* que pode incluir todos os objetos reordenados (permutação); ou representação *selection based* que pode representar apenas mochilas válidas. No primeiro caso, necessita-se de alguma heurística para validar as mochilas.

B. Tratamento de Restrições

A maioria dos métodos apresentados necessita de algum modo de induzir, manter ou gerar mochilas válidas. O método mais comum de indicação de soluções viáveis é a função de penalidade, embora alguns procedimentos utilizem preferências. Já para manter ou gerar mochilas válidas é necessária a aplicação de algumas heurísticas [Hinterding1999].

- Funções de penalidade – são usadas para diminuir o valor de *fitness* de determinada solução que viola alguma restrição, neste caso a capacidade da mochila. A dificuldade é determinar de maneira precisa os coeficientes dessa função além de que as próprias

soluções inviáveis podem conter informações que encaminham o algoritmo dentro do espaço de busca [Michalewicz1996].

- Preferências – também é outra maneira de encorajar a busca de soluções viáveis. Em vez de penalizar as soluções inviáveis, adotam-se regras que permitam que nenhuma solução inviável apresente um *fitness* melhor que outra solução viável. Assim, enquanto as viáveis são comparadas pelo *fitness*, as inviáveis são comparadas pela violação. Este método combina as idéias das pesquisas de Hinterding [Hinterding1998] e Fonseca [Fonseca1998a];
- Heurísticas – são métodos utilizados para manter ou gerar mochilas válidas. Elas podem ser usadas como parte de um decodificador ou como parte de um operador de cruzamento especializado. Em ambas as situações a tarefa é converter uma mochila inválida noutra válida (*i.e.* satisfazer a restrição que foi violada). Existem diversas heurísticas simples que realizam esse trabalho. Hinterding listou algumas divididas em dois grupos [Hinterding1999]: (i) existem as heurísticas dependentes da ordem dos objetos no cromossomo, tais como *Next fit* e *First Fit*; e as que independem da ordem, ou seja, avaliam o conjunto de objetos selecionados, *e.g.* *Best Fit* e *First Fit Descending*.

2.4 Redes Neurais Artificiais

Redes neurais artificiais (RNA) foram inspiradas, no funcionamento do sistema nervoso humano. Ou seja, tenta-se reproduzir no computador o modo como o cérebro processa informações com o objetivo de incluir características necessárias para a resolução de problemas complexos e com alto volume de informações existentes no mundo real. Essas qualidades adquiridas com inspiração biológica e acrescidas à computação tradicional transformam o computador numa plataforma que (i) se adapta facilmente aos problemas; (ii) apresenta alto grau de paralelismo; (iii) tem capacidade de generalização e (iv) robustez.

Até mesmo cérebros primitivos e pequenos são capazes de realizar atividades complexas em fração de segundos, por exemplo, a visão de um pássaro ou o olfato de um cão. Estas mesmas atividades levariam muito mais tempo para serem realizadas pelos computadores atuais (utilizando computação tradicional).

O conhecimento é adquirido pelo cérebro através da vivência, da experimentação. Toda essa experiência é acumulada ao longo do tempo por estruturas celulares chamadas neurônios

especificamente sob a forma de ligações sinápticas. Os mecanismos de ajuste dos valores (*i.e* pesos sinápticos) materializam o processo de aprendizagem.

McCulloch e Pitts desenvolveram um neurônio artificial que é uma abstração do neurônio real. O neurônio MCP, como ficou conhecido, possuía entradas x_1, x_2, \dots, x_p que eram potencializadas pelos pesos sinápticos w_1, w_2, \dots, w_p e processadas (somatório) pelo neurônio. A resposta y do neurônio é o resultado da avaliação de uma função de ativação, que dependendo de um limiar pode ou não ativar o neurônio [McCulloch1943]. A Figura 2.14 ilustra o neurônio artificial. A partir deste neurônio, desenvolve-se o modelo matemático de uma RNA que tenta emular a atividade cerebral.

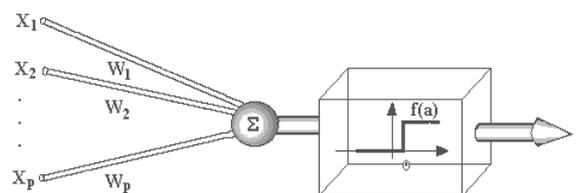


Figura 2.14. Neurônio artificial MCP desenvolvido por McCulloch e Pitts [McCulloch1943].

2.4.1 Definição de RNA

A organização destes neurônios artificiais conectados por ligações sinápticas em variados formatos permite ao computador armazenar informações e gerar novos conhecimentos.

Haykin define uma rede neural como uma máquina adaptativa [Haykin1994]:

“Uma rede neural é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. Ela se assemelha ao cérebro em dois aspectos:

- i. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.*
 - ii. Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.”*
- ([Haykin1994], pág. 28).

A unidade básica de processamento de uma RNA é o neurônio artificial que é composto pelos seguintes elementos:

- Um conjunto de sinapses ou elos de conexão, representando as entradas do sistema com pesos próprios;

- Um somador, que processa as entradas ponderadas com os respectivos pesos realizando uma combinação linear;
- E, uma função de ativação ou função restritiva, que limita a amplitude de saída de um neurônio. Existem vários tipos de função de ativação, os mais conhecidos são: função de limiar, função linear por partes e função sigmóide (logística ou tangente hiperbólica).

2.4.2 Processos de Aprendizagem

A capacidade de aprendizagem é uma das mais importantes características de uma RNA. Ela é responsável pela adaptação, dinamicidade e aplicabilidade das redes neurais aos problemas do mundo real. Sem ela, a rede não aprenderia aspectos relevantes do ambiente e tão pouco seria capaz de melhorar seu desempenho ao longo do tempo.

O termo aprendizagem pode ser empregado em várias áreas, com diversas interpretações. Neste trabalho, usaremos a seguinte definição:

“Aprendizagem é um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual está inserida. O tipo de aprendizagem é determinado pela maneira como a modificação dos parâmetros ocorre.” ([Haykin1994], pág. 75).

Com relação aos paradigmas de aprendizagem, estes podem variar de acordo com o tipo de RNA e, principalmente, de acordo com a natureza dos dados a serem processados. Eles se dividem em duas grandes categorias:

- Supervisionado – quando há disponibilidade de uma base de dados com entradas e saídas. A saída produzida pela RNA é comparada com a saída esperada - fornecida pela base de dados. Em seguida, calcula-se o erro obtido por fim, executa-se um procedimento de reajuste dos pesos sinápticos objetivando produzir, nas próximas iterações, valores de saída na rede mais aproximados dos esperados. Em geral, a aplicação deste método possibilita uma convergência (estabilidade em relação ao ajuste de pesos) mais rápida. Entretanto, obter os dados para o treinamento da rede (que servem como “professor” da rede) pode ser uma tarefa difícil dependendo do domínio do problema. Outro fator a considerar é a representatividade da base de dados em relação ao problema, o que pode comprometer futuras generalizações;
- Não supervisionado – nesta segunda categoria não há presença de um “professor” que indique o resultado esperado e possibilite o cálculo de qualquer erro. A RNA tem que capturar do ambiente as informações necessárias para que os ajustem dos pesos

levem à solução. Este tipo de aprendizado pode ser subdividido em dois grupos: (i) o aprendizado por reforço; e (ii) o aprendizado não-supervisionado. No primeiro existe a presença de um *crítico* que serve como balizador do treinamento. Apesar de não informar o erro precisamente, pode indicar algum caminho a ser seguido. Já na segunda abordagem, a rede manipula os dados de forma a extrair conhecimento a partir da determinação de propriedades deste conjunto. Pode-se utilizar a regra de aprendizagem competitiva na implementação do aprendizado não-supervisionado. Os neurônios de uma camada escondida competem entre si na tentativa de solucionar o problema. As principais redes baseadas neste paradigma são: os mapas auto-organizáveis (do inglês, *self organized maps*, SOM) de Kohonen [Kohonen1995] e os modelos ART de Grossberg e Carpenter [Carpenter1988].

Por fim, acrescenta-se que não existe um paradigma ideal de aprendizagem, sua escolha depende da natureza dos dados, do problema, e da abordagem escolhida para a arquitetura e classe da rede neural.

2.4.3 Redes *Multi-Layer Perceptron*

Após a publicação de Rosenblatt sobre aplicações de computação com aprendizado na década de 60 [Rosenblatt1961], permeava na comunidade acadêmica uma grande euforia a cerca de redes neurais artificiais. Nesta época, comprovou-se o teorema da convergência do *Perceptron* o que prometia uma revolução nos sistemas computacionais. No entanto, no final da década de 60, Minsky e Pappert demonstraram que apesar da convergência, o *Perceptron* não era capaz de resolver problemas relativamente simples [Minsky1969]; ele só tratava de problemas linearmente separáveis. Com isso, muitos passaram a não ver mais RNAs como uma ferramenta efetiva para resolver problemas reais e, conseqüentemente, as pesquisas foram quase que abandonadas.

Nos anos seguintes poucas pesquisas foram realizadas. O *Perceptron* só teria poder se fosse encadeado em forma de camadas, mas ainda não existiam técnicas eficazes para operacionalizar esse tipo de aprendizado. Finalmente, na década de 80, Rumelhart *et al.* [Rumelhart1986] descreve um algoritmo capaz de “ensinar” as redes *perceptron* de múltiplas camadas (do inglês, *Multi-Layer Perceptron*, MLP). Na verdade, desde o início da década de 70 esta solução já tinha sido concebida, mas pela pouca divulgação sua contribuição não havia sido aplicada [Werbos1974]. Esse algoritmo foi chamado de retropropagação do erro (do inglês, *Error Backpropagation*), pois seu princípio é propagar o erro desde a camada saída até a camada de

entrada, ou seja, em cada camada os pesos sinápticos de cada neurônio são compensados de acordo com sua relevância para o erro produzido [Jain1996].

A partir de então, o entusiasmo retornou e as pesquisas estão sempre mantidas em grande volume na área. Já comprovadas como aproximadoras universais de funções, as redes MLP são uma técnica revolucionária de resolver problemas complexos.

Por emularem o cérebro humano em operações inteligentes, as RNA MLP são capazes de reconhecer padrões, realizar otimizações, classificações, regressões e previsões. De modo que já foram aplicadas com sucesso nas mais diversas áreas de conhecimento: na classificação de câncer [Fi-Liu2004], no tratamento de câncer urológico [Abbod2007], na previsão de refluxo em crianças [Knudson2007], em análise financeira [Huang2007], na previsão de regimes (dietas) adequados [Hu2007], no combate à poluição ambiental [Duque2007], no reconhecimento de imagens [Sinha1998], na agricultura [Park2004] e em particular, no problema da colheita da cana-de-açúcar [LimaNeto1998] [Pacheco2005] [Pacheco2006] [Oliveira2006] [Pacheco2007].

A. Parametrização Inicial da Rede MLP

Apesar das constantes pesquisas, as redes MLP revelam certa dificuldade em seu uso. A grande quantidade de parâmetros configuráveis exige do usuário experiência, que provavelmente apenas será adquirida através de muita utilização prática. Ainda não existem estudos que comprovem com precisão como o ajuste paramétrico e a topologia da rede devem ser definidos.

Dentre os desafios a contornar na parametrização das RNA destacamos: modelagem do problema (o domínio do problema e características das entradas); a definição da topologia da rede; o número de camadas escondidas; o número de neurônios nessas camadas; e o número de conexões entre essas camadas. Outro procedimento que influencia o treinamento de uma RNA é a inicialização dos pesos da rede. Geralmente são atribuídos valores aleatórios entre -1 e 1. Simulações procedidas por redes inicializadas com valores distintos produzem saídas distintas também. A definição da taxa de aprendizado e condições de treinamento também são parâmetros de difícil definição, vide seção a seguir.

B. Treinamento

Com a rede configurada e os dados preparados para o processamento (divididos entre entradas e saídas desejadas), inicia-se a fase de treinamento. Esta é a fase mais importante e mais prolongada no uso das MLP. Pois, treinar uma rede neural é torná-la apta para processar e resolver o problema para o qual foi planejada. Nas redes MLP o treinamento em si é composto

por duas etapas: propagação dos dados de entrada (*forward*) e retropropagação dos erros de saída (*backward*).

Durante o treinamento das redes MLP, deve-se decidir quantos padrões farão parte do conjunto de treinamento e quantos serão para testes. Esse é um *trade-off* importante, pois quanto maior for o conjunto de treinamento, mais chances sua rede neural terá de generalizar bem os aspectos do problema. Por outro lado, se não existe um conjunto de teste representativo para o problema, não há como avaliar o treinamento nem medir a precisão da rede MLP (esperada) em suas futuras aplicações. Assim, cabe o bom senso na divisão dos dados entre treinamento e teste.

Na seqüência determina-se diversos parâmetros de treinamento, entre eles destacam-se:

- Número de ciclos (épocas de treinamento) – define a quantidade de vezes que todo o conjunto de padrões será apresentado à rede neural durante o treinamento (uma época). Este número está intimamente ligado à taxa de aprendizado. Dois problemas atrelados à escolha do parâmetro são *underfitting* e *overfitting*. O primeiro cuja tradução aproximada é “sub-ajuste” ocorre quando a rede não é treinada suficientemente para se tornar capaz de produzir resultados satisfatórios, convergir; e o segundo cuja tradução aproximada é “super-ajuste” ocorre quando a rede é treinada demasiadamente e passa a decorar os dados apresentados, *i.e.* perde a capacidade de generalização. Isso ocorre quando o modelo incorpora ruídos ou inconsistências atreladas ao conjunto de treinamento [Principe2000];
- Taxa de aprendizado ou η (*eta*) – define a velocidade do aprendizado, ou seja, a velocidade em que os erros serão corrigidos (veja um exemplo na Figura 2.15). Uma taxa de aprendizado alta representa um grande ajuste nos pesos sinápticos, *i.e.* rapidamente a função caminhará na direção do menor erro. No entanto, é possível que a função nunca encontre um erro aceitável nem consiga convergir justamente devido a esse grande ajuste. A Figura 2.16 ilustra esse fenômeno. Se relacionarmos a taxa de aprendizado com o número de ciclos em um treinamento, pode-se dizer que estes dois parâmetros possuem grandezas inversamente proporcionais, pois se for usado uma taxa de aprendizado baixa, a rede neural convergirá lentamente na direção do erro mínimo e por isso necessitará de um número maior de ciclos e vice-versa [Principe2000];

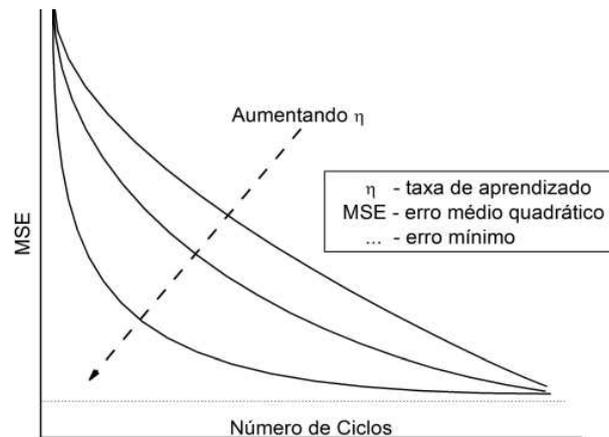


Figura 2.15. Diferentes taxa de aprendizado e suas respectivas curvas de aprendizado. Adaptada de [Principe2000].

A Figura 2.16 mostra claramente a variação da taxa de aprendizado. Nela podemos observar que quando é escolhida uma taxa pequena, lentamente a rede caminhará na direção do erro mínimo. Quando é escolhida uma taxa um pouco maior a rede consegue encontrar um erro aceitável, mais rapidamente, mas oscila em torno do erro mínimo. No entanto, se a taxa de aprendizado for muito grande, a rede neural pode nunca conseguir convergir.

A fase de retropropagação do erro é iniciada logo após a propagação e geração do erro, com a chamada do algoritmo *backpropagation*. Este implementa a correção dos pesos sinápticos de todos os neurônios através da regra do gradiente descendente que computa a derivada parcial das saídas de cada neurônio em relação aos pesos sinápticos atuais. Este ciclo de entrada de dados e correção dos pesos é justamente o processo de aprendizagem, que mencionamos anteriormente.

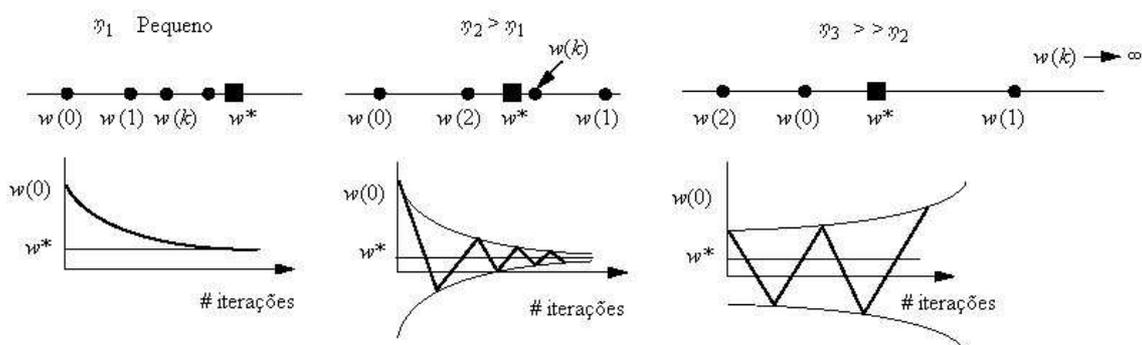


Figura 2.16. Relação entre taxa de aprendizado e convergência a partir do ajuste de pesos, extraída de [Principe2000].

Este ciclo se repete até o final do treinamento, no entanto, é necessário definir quais serão os critérios de parada do treinamento. Como afirmado anteriormente, se o número de ciclos for o

único indicador de término de um treinamento, dois grandes riscos surgem: a rede neural pode sofrer um *underfitting* ou *overfitting*.

Existem outras abordagens de critérios de parada que não apenas o número de épocas. Um primeiro poderia simplesmente ignorar o número de ciclos e definir uma meta de erro, *i.e.* não termine o treinamento enquanto o erro estiver acima de 3%, por exemplo. Certamente, esta abordagem também não é muito aconselhada, pois impor uma meta fixa, pode levar a rede a sofrer de *overfitting*. Outra abordagem bastante popular para critério de parada é a GL5 [Prechelt1994]. Nele, divide-se a base de dados em treinamento, testes e em um conjunto adicional utilizado na validação do treinamento. O tamanho deste novo conjunto deve ser uma fração do conjunto de treinamento. O procedimento funciona da seguinte maneira:

1. A cada fração de padrões de treinamento apresentados à MLP, insere-se um padrão do conjunto de validação;
2. Calcula-se o erro obtido na propagação do padrão de validação;
3. Guarda-se o menor erro obtido, bem com o estado da RNA respectivo (valores dos pesos sinápticos);
4. Compara-se o erro calculado com o menor erro que foi guardado.
5. Se o novo erro for 5% maior que o menor erro – o treinamento é parado, pois a rede neural poderia estar perdendo sua generalização, e a configuração da melhor rede é restaurada. A Figura 2.17 mostra o melhor momento para finalizar o treinamento;
6. Se o novo erro for menor que o menor erro já encontrado, realiza-se o passo 3 e continua-se o treinamento repetindo os passos acima;
7. Senão (passo 5 e 6 falsos) nada é modificado e o treinamento continua até quando um dos critérios (número de épocas ou passo 5) seja satisfeito.

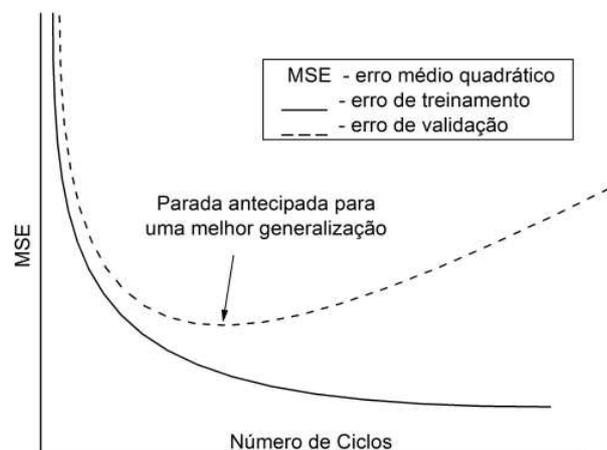


Figura 2.17. MSE de treinamento e validação – critério de parada GL5.

Posteriormente, deve-se escolher qual será o critério para medir o desempenho da rede. Em geral, esta decisão é dependente do problema, por exemplo: quando se lida com problemas de classificação, a avaliação da rede é feita pela taxa de classificação enquanto que em problemas de regressão o erro médio quadrático é a medida utilizada.

Com o término do treinamento, deve-se apresentar o conjunto de testes à rede encontrada. Esse conjunto realizará apenas a fase *forward* (que não altera nenhum peso) e ao final obtém-se o erro da MLP. Observe que existem também os erros de treinamento e de validação, mas o erro real para esta rede MLP é o calculado a partir do conjunto de testes, pois é este conjunto que possui os padrões que nunca foram vistos pela RNA nem durante o treinamento e nem na validação. Mais uma vez a análise dos resultados (do erro obtido) está intimamente ligada ao problema. Por exemplo, em um sistema baseado em redes neurais artificiais que propõe o controle de acesso a contas bancárias através do reconhecimento de voz, pode-se exigir um erro mínimo (talvez, abaixo de 1%), já em um outro que faz previsões sobre melhores rotas de trânsito pode-se admitir um erro de 20%.

Na Engenharia das redes neurais artificiais – em especial no uso de redes *Multi-Layer Perceptron*, poucas regras apóiam os engenheiros nas suas decisões de projetos. Portanto o experimentalismo contextualizado é uma das poucas maneiras de se conceber e implementar soluções com redes neurais artificiais. Mesmo assim, com muitas decisões heurísticas, RNA são ferramentas poderosas para sistemas de decisão. Nesta proposta RNA são utilizadas como mecanismos na geração de valores futuros para as variáveis do problema da colheita da cana-de-açúcar (indicadores agrônômicos de produtividade).

Capítulo 3

Domínio de Aplicação

Neste capítulo focamos nos conceitos relacionados com o domínio de aplicação deste trabalho, *i.e.* (i) cana-de-açúcar e (ii) sistemas de suporte à decisão. Descrevemos ainda alguns trabalhos relacionados entre estes dois temas e indicamos como a pesquisa proposta dá continuidade a outras anteriores.

3.1 Cana-de-açúcar

3.1.1 História

A cana-de-açúcar é utilizada pelo homem há milhares de anos. É talvez, o único produto agrícola destinado à alimentação que ao longo da história mobilizou homens e nações e foi alvo de disputas e conquistas em larga escala [Machado2008]. Sua provável origem é de regiões próximas à Índia há cerca de 8 mil anos e foi do sânscrito (antiga linguagem indiana) *sarkara* que originaram-se às versões indo-européias da palavra açúcar: *sukkar* em árabe, *saccharum* em latim, *zuccherò* em italiano, *seker* em turco, *zucker* em alemão, *sugar* em inglês [Copersucar2008].

As primeiras observações da cana-de-açúcar por ocidentais ocorreram durante as conquistas de Alexandre, O Grande (em torno de 325 a.C.). Os persas e árabes foram os responsáveis por sua disseminação no Egito, Síria, Grécia, até a Sicília. Também foram os persas que descobriram o processo de refinamento do açúcar o que impulsionou esta atividade comercial (por volta de 600 d.C.). Os mulçumanos, após invadirem a Pérsia aprenderam a fazer o açúcar sólido e iniciaram o plantio da cana em todos os seus territórios, inclusive em Portugal e na Espanha [Agroclubes2008].

Portugal foi pioneiro na produção extensiva, o que subsidiou as grandes conquistas marítimas. Com os novos descobrimentos, a cana-de-açúcar foi trazida para as Américas. Acredita-se que Colombo iniciou o cultivo da cana onde hoje é a República Dominicana. No Brasil, a primeira muda de cana-de-açúcar foi oficialmente plantada por Martim Affonso de Souza na Capitania de São Vicente em 1532, mas foram notadamente nas Capitanias de Pernambuco e Bahia que a produção da cana foi mais acentuada [Agroclubes2008].

Em poucos anos o Brasil passou a monopolizar a produção mundial de açúcar com uma lucratividade bastante elevada para Portugal e Holanda. No entanto, em 1578 com a incorporação de Portugal pela fervorosa católica Espanha, a protestante Holanda foi banida das negociações. Depois de anos de guerras e invasões os holandeses iniciaram sua produção no Caribe, seguidos pelos ingleses e franceses. Era o fim do monopólio brasileiro. A concorrência continuou aumentando até o século XVIII. No início do século XIX, graças ao bloqueio continental aplicado pela Inglaterra sob Napoleão, a Europa passou a produzir açúcar a partir de beterraba e não dependia mais de importações. Com a Revolução Industrial, as produções a partir da beterraba e da cana-de-açúcar foram significativamente elevadas e o modelo escravista de produção aos poucos foi abandonado [Machado2008].

Após as grandes guerras do século XX, o Brasil retoma sua posição de grande produtor de cana. Mais recentemente, no final do século, com o agravamento da segunda crise do petróleo e programas governamentais brasileiros como o Próalcool, a cana passou também a ser representativa fonte de combustível renovável produzindo o etanol. Atualmente, a indústria sucro-alcooleira continua em forte expansão tendo como seus principais produtos: o açúcar, o álcool e mais recentemente, a energia (disponibilizada a partir da queima de sua biomassa).

3.1.2 Botânica e Fisiologia

A cana-de-açúcar é uma planta do gênero *Saccharum* e é da mesma família (*Poaceae*) do milho, sorgo, arroz, entre outras gramíneas [Agroclubes2008]. É uma gramínea semiperene de grande porte que armazena altas taxas de sacarose em seus colmos suculentos. Por ser da classe das *Monocotiledôneas*, pode se reproduzir através de brotamento, ou seja, mesmo após vários cortes a planta refloresce utilizando a mesma base radicular, descartando a necessidade de replantios anuais [Aranha1987]. Apresenta várias espécies híbridas e modificadas geneticamente para se tornarem mais resistentes a pragas ou mais produtivas, por exemplo.

É uma cultura pouco exigente de solo, no entanto alcança melhores produtividades em terrenos com boa aeração e drenagem, e com profundidade superior a um metro

[Copersucar1988]. O primeiro ciclo da cultura pode durar 12 ou 18 meses, dependendo da espécie e são respectivamente conhecidas como *cana de ano* e *cana de ano e meio*. A partir do primeiro corte e durante os brotamentos seguintes o vegetal passa a ser conhecido como *cana soca*. Para que as safras de cana soca (brotamentos) atinjam boa produtividade é necessário realizar investimentos em insumos e tratamentos culturais [Picoli2006].

A cana-de-açúcar se adaptou muito bem ao Brasil, em especial ao clima. O vegetal possui grande capacidade fotossintética e, portanto a radiação solar afeta diretamente todo desenvolvimento da cultura. Em regiões com pouca luminosidade, as canas apresentam colmos finos e longos o que dificulta o acúmulo de matéria [Fortes2003]. Também por causa do clima, o Brasil apresenta duas safras distintas, uma no Nordeste (de novembro a abril) e outra na região Centro-Sul (de abril a novembro).

O ciclo de vida da cana-de-açúcar pode ser dividido em três momentos: crescimento, maturação e reprodução. Durante o crescimento do vegetal o teor de sacarose é muito baixo, pois toda energia é utilizada no intenso desenvolvimento da cana. Com o início da seca, a cana-de-açúcar realiza basicamente acúmulos energéticos, ou seja, o teor de sacarose chegará em seu limite máximo – esta é a fase de maturação. Por fim, se não colhida, a cana começa sua reprodução [Magalhães1987]. O final da maturação é o momento ideal para proceder à colheita, pois a cana estará com seus açúcares mais elevados. A partir daí, a produtividade da colheita tende a ser reduzida, pois o vegetal inicia o processo de reprodução utilizando a energia (açúcares) que tinha acumulado [Machado1987].

A maturação é um processo fisiológico que é regido por vários fatores naturais, tais como: variedade da cana, clima, tipo de solo e tratamentos culturais. Diferentes variedades apresentam diferentes teores de sacarose mesmo quando cultivadas sob as mesmas condições [Picoli2006]. Assim, determinar o momento em que a cana-de-açúcar está com seus açúcares mais elevados (final da maturação e antes da reprodução) é uma atividade complexa e não determinística.

3.1.3 O Processo de Colheita

A colheita da cana-de-açúcar é uma das atividades mais importantes da cultura, pois influencia diretamente nos resultados obtidos, ou seja, nos resultados econômicos produzidos pela safra. A colheita deve ser um processo contínuo, pois a cana que é moída retorna automaticamente ao sistema sob a forma de bagaço, alimentando as caldeiras da própria usina. Qualquer interrupção no processo implica em prejuízos.

Cabe aos engenheiros agrônomos decidir quando e quais lotes de cana serão colhidos. Essa escolha deve considerar diversos fatores relacionados com a atividade econômica, por exemplo: os lotes selecionados devem estar em seus momentos de maturação máxima para que o aproveitamento agrônômico do canavial seja maximizado; os custos operacionais com mão-de-obra, maquinário, transporte e produção devem ser minimizados para viabilizarem a atividade; e as diversas restrições do problema devem ser obedecidas, tais como: capacidade de transporte, capacidade de moagem, número de frentes de corte, distância do lote à usina, entre outras.

As decisões de colheitas podem ser avaliadas a partir de alguns indicadores de produtividade, dos quais destacam-se: PCC, ATR, FIBRA e TCH. Estes já foram definidos no primeiro capítulo.

3.1.4 Fatores Determinantes de Produtividade

Vários são os fatores que influenciam na produtividade da cana-de-açúcar. Com o objetivo de determinar a representatividade dos fatores na produtividade agrícola, Lima Neto os classificou em duas categorias [LimaNeto1998]:

- Fatores usuais – são aqueles que ocorrem para toda a amostra considerada e seus efeitos podem ser medidos e atribuídos individualmente ou coletivamente. Os fatores usuais podem ainda ser subdivididos em três subcategorias:
 - Fatores ambientais tais como: temperatura, intensidade de luz, disponibilidade de água e nutrientes, umidade do ar, entre outros;
 - Fatores intrínsecos aos lotes como: variedade da cana, estágio de maturação, tipo do solo, data do plantio, data do corte, etc;
 - Fatores logísticos tais como: área do lote, distância até a unidade de moagem, capacidade de transporte, tempo médio de transporte, disponibilidade de sistema viário, entre outros.
- Fatores eventuais – não ocorrem em toda amostra, mas sua influência é significativa e deve ser considerada (se possível). Entre eles destacam-se: compactação do solo, déficit hídrico, macro-nutrientes, correções de pH, irrigação, pragas, incêndios, maturação artificial, etc.

3.1.5 Importância Econômica da Cultura Canavieira

O Brasil é o maior produtor mundial de cana-de-açúcar seguido pela Índia, China, Tailândia e Paquistão. A Figura 3.1 ilustra os 10 maiores produtores de cana do mundo e mostra que o Brasil

sozinho, consegue produzir mais que os 4 produtores seguintes juntos [FAO2007]. Na safra de 2006/2007 produziu mais de 428 milhões de toneladas de cana-de-açúcar, o que resultou em 30 milhões de toneladas de açúcar e 17 milhões de litros de álcool [MinisterioAgricultura2008a]. A Figura 3.2 mostra como a produção brasileira vem aumentando nos últimos anos – desde 2001 cresceu mais de 50%. O setor canavieiro brasileiro é responsável por mais de 1 milhão de empregos diretos, onde metade destes apenas na produção da cana-de-açúcar e a outra metade distribuída na agroindústria do açúcar e do álcool [Picoli2006]. O agronegócio brasileiro é responsável por 33% do Produto Interno Bruto (PIB) brasileiro, 42% das exportações totais e 37% dos empregos totais do país [MinistérioAgricultura2008b]. A cana-de-açúcar reúne cerca de 6% dos empregos agroindustriais de todo o Brasil [UNICA2006].

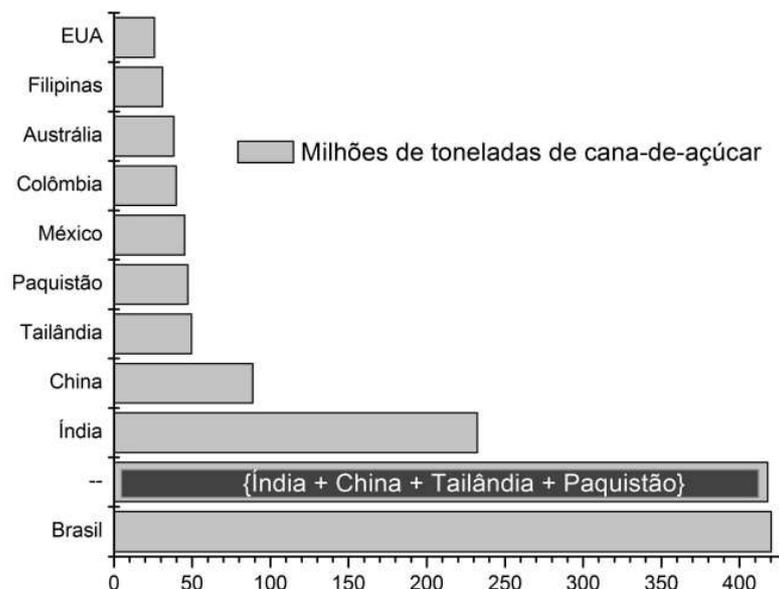


Figura 3.1. Lista dos 10 maiores países produtores de cana-de-açúcar-de-açúcar do mundo [FAO2007].

Um dos fatores responsáveis pelo aumento do interesse sobre a cana-de-açúcar, em especial sobre o etanol é o protocolo de Quioto, assinado em 1997 por vários países membros das Nações Unidas. Neste protocolo, os países se comprometeram em reduzir as emissões de gases causadores do efeito estufa entre 2008 e 2012. Uma das alternativas para esta redução é a mistura de álcool na gasolina o que minimiza os efeitos nocivos da octanagem produzida pelos combustíveis fósseis [Picoli2006]. Além de fatores ambientais, fatores econômicos como a alta instabilidade do preço do barril de petróleo também contribui para um aumento na procura do etanol [Figueira2005].

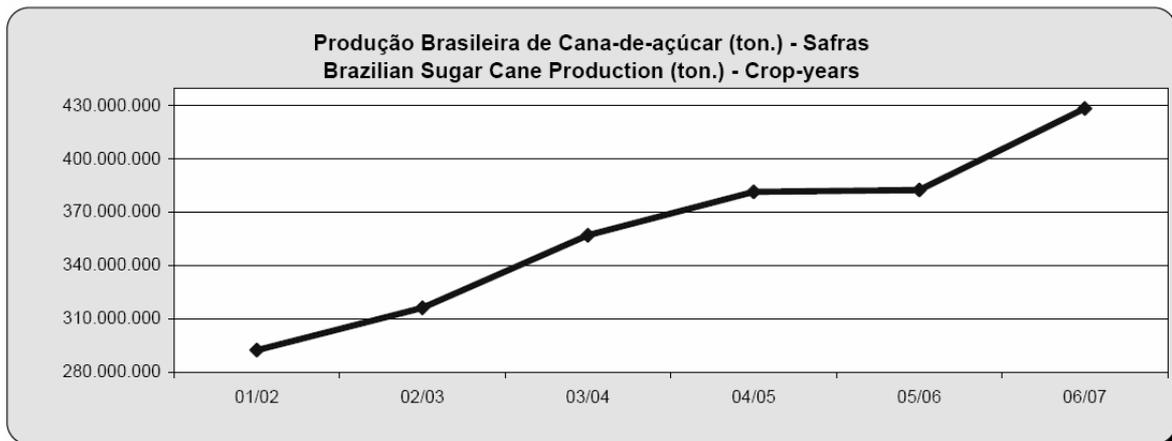


Figura 3.2. Evolução da produção de cana-de-açúcar no Brasil entre os anos de 2001 e 2007. Extraído de [MinisterioAgricultura2008a].

Por fim, um terceiro produto vem revolucionando a indústria canavieira – a produção de energia. Segundo a Empresa de Pesquisa Energética (EPE), pela primeira vez na história, a cana-de-açúcar ultrapassou as hidrelétricas na matriz energética brasileira em 2007. Os derivados da cana responderam por 16% da matriz contra os 14,7% das hidrelétricas, mas a liderança continua com o petróleo e seus derivados com 36,7%. No entanto o destaque é que 70% das energias obtidas foram de fontes renováveis em 2007 com o destaque especial para a cana-de-açúcar. De acordo com Maurício Tolmasquim, presidente da EPE, a cana vai se manter como parte importante da matriz energética e ele ainda acredita que esta tendência seja irreversível [IBPS2008].

3.2 Suporte à Decisão Gerencial

Nos primórdios da computação eletrônica as aplicações eram isoladas e independentes, sem uma preocupação real com duplicação de dados e racionalização do processamento. A partir de década de 60, informação e tecnologia passaram a andar juntas. O avanço tecnológico evidenciou a necessidade de uniformidade e integração das informações como forma de organização e alcance dos objetivos - era o nascimento dos sistemas de informação [Rezende2003] [Pacheco2006].

Os sistemas de informação gerencial (SIG, do inglês *management information system* – MIS) surgiram como produtos de informação que apoiavam as decisões estruturadas. Eles eram destinados ao controle operacional, exibindo, basicamente, relatórios pré-definidos. No início da década de 70, começaram a surgir outros tipos de sistemas, os sistemas de suporte à decisão (SSD, do inglês *decision support system* – DSS). Diferente dos SIG, os SSD permeavam numa economia instável, onde melhores informações são uma vantagem competitiva importante. Assim

muitas empresas e centros de pesquisas passaram a desenvolver SSD [Sprague1991]. No entanto, o processo decisório continua, ainda hoje, sendo um dos mais desafiadores papéis de um gerente [Pacheco2006].

3.2.1 Tipos de Sistemas de Informação

Os sistemas de informação (SI) podem ser classificados de várias formas. Não existe uma regra que defina se um sistema é de um determinado tipo ou de outro, até porque um SI pode ser reconhecido em mais de uma categoria. Diversos autores criaram classificações diferentes, cada um a partir de suas perspectivas: (i) Zaneti com relação ao escopo dos sistemas [ZanetiJr2003]; (ii) Lucas através do tipo de decisão que suportam e o nível hierárquico a que se destinam [Lucas1997]; e (iii) O'Brien a partir das funções que os sistemas desempenham [O'Brien2001]. A Figura 3.3 ilustra esta última classificação.

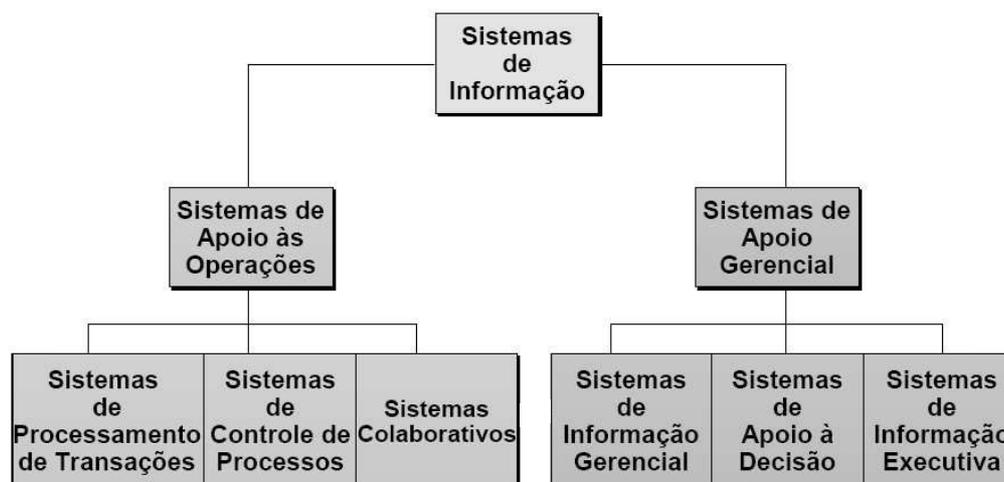


Figura 3.3. Classificação de sistemas de informação de acordo com O'Brien [O'Brien2001].

3.2.2 Tomada de Decisão e Solução de Problemas

Sabe-se que para qualquer tipo de organização empresarial existe uma série de decisões a serem tomadas. A eficácia e a eficiência dessas decisões é que vão tornar claros os objetivos intencionais dos seus processos de valor adicionado e, conseqüentemente, delineiam a cultura da organização com os tipos e as abordagens de decisões. Comumente o planejamento estratégico e as metas globais de uma empresa são os balizadores que definem o estágio para adoção dos processos de valor adicionado e de tomada de decisão. Estes, geralmente são subsidiados pelos sistemas de informação destas organizações [Stair1999].

Tendo em vista a variedade e a complexidade dos problemas que surgem nas empresas é que emerge a importância do processo de tomada de decisão. Estes problemas podem variar dos mais simples até os mais complexos, e comumente são advindos de fatores externos. Assim, para solucionar estes problemas as empresas comumente aplicam duas abordagens diferentes: (i) a reativa, onde o solucionador espera que o problema de fato ocorra para buscar a solução; e (ii) a pro-ativa, onde o solucionador, numa atitude preventiva, antecipa-se aos problemas antes que qualquer medida reativa seja necessária. Cabe aos solucionadores definir qual o melhor momento para a utilização de tais abordagens [Stair1999].

A. Tomada de Decisão como Componente da Solução de um Problema

Solucionar problemas em uma organização empresarial ainda é o grande e maior desafio. O processo de solucionar estes, definido por George Huber, é composto por três passos: a tomada de decisão, a implementação e o monitoramento. Em particular, Herbert Simon dividiu a fase de tomada de decisão em três outras partes: inteligência, projeto e escolha [Simon1960]. Assim o primeiro passo para solucionar problemas é a tomada de decisão e exige do decisor competência, inteligência e determinação para escolher qual a melhor opção a ser tomada observando a possível dinamicidade deste processo, sem deixar de investigar a causa e o escopo da atividade [Laudon2000].

Diversos estudos e pesquisas apontam a viabilidade dos sistemas de informação para determinar, identificar, explorar e analisar alternativas dando assistência na seleção final e no monitoramento da implementação da decisão [Turban2006] [McNurlin2005] [Stair2005].

B. Decisões Programadas x Decisões Não Programadas

É na etapa de escolha que os tomadores de decisão (DM, do inglês *decision maker*) são apresentados aos muitos fatores determinantes para decisões de problemas. Um destes fatores é se a decisão pode ser programada ou não. Decisões programadas são tomadas de acordo com regras, um procedimento ou um método quantitativo. Logo, são facilmente implementadas utilizando-se sistemas de informação tradicionais. Os SIG também são usados para resolver decisões programadas [Stair2005].

Por outro lado, as decisões não programadas lidam com situações mais complexas, incomuns ou excepcionais, sendo em muitos casos difíceis de se quantificar tendo em vista a singularidade de cada situação. Assim, não cabe a aplicação de regras ou de procedimentos previamente determinados. Os sistemas de apoio à decisão e os sistemas especialistas têm sido os

mais requisitados para estas situações, onde os problemas são semi-estruturados ou não estruturados [Turban2006].

C. Grau de Risco

Para os DM outro fator complicador da fase de escolha é o grau de risco que suas escolhas irão proporcionar. Eles devem avaliar previamente qual a probabilidade de sucesso ou não da suas soluções (escolhas) na resolução dos problemas. Existem diversos tipos diferenciados de risco nos ambientes de tomada de decisão: (i) a tomada de decisão sob certeza, onde já se sabe o que vai acontecer, pois todos os passos são conhecidos e esperados pelo DM, e (ii) a tomada de decisão sob incerteza, onde só existem estimativas prováveis para o seu resultado e não há garantias do produto esperado [McNurlin2005].

Na tomada de decisão sob incerteza existem dois tipos de risco: (i) o risco da decisão e (ii) o risco da estimativa. O primeiro é inerente à própria decisão e não pode ser modificado enquanto o segundo se refere ao próprio processo de estimativa. Talvez o principal aspecto de decisões sob incerteza seja conseguir minimizar os riscos de avaliação. Isto requer do tomador de decisão avaliações criteriosas sobre quais fatores são de fato relevantes e conseguem aproximar as estimativas à realidade. Vale salientar que os modelos computadorizados que lidam com problemas de tomada de decisão sob incerteza e os sistemas de informação não podem eliminar totalmente o risco [Stair1999].

D. Abordagens de Otimização, Satisfação e Heurística

Os sistemas computadorizados de apoio à decisão podem tanto otimizar quanto satisfazer as necessidades e interesses dos usuários. Os modelos de otimização utilizam-se dos obstáculos do problema, enquanto os de satisfação utilizam normalmente da solução com maior probabilidade de dar bons resultados, ou seja, não se busca necessariamente a melhor solução, mas sim uma que seja satisfatória para o problema [McNurlin2005].

O modelo de satisfação é considerado um bom método alternativo de modelagem, porque às vezes, é bastante oneroso analisar todas as alternativas até se alcançar a melhor solução. Já os modelos de otimização são mais adequados para problemas que podem ser modelados matematicamente e com baixo grau de risco de avaliação. Quanto ao método heurístico (*i.e.* empírico) são diretrizes ou procedimentos preconizados que geralmente encontram uma boa solução e são bastante utilizados no processo de tomada de decisão [Laudon2000].

Em todas as abordagens existem inúmeros fatores condicionantes para a solução de problemas, os quais, se bem assimilados pelo administrador, aumentarão sua capacidade de análise do problema e, conseqüentemente, de buscar as melhores decisões. Tais fatores incluem objetivos de decisão, alternativas aumentadas, competição, criatividade, ações sociais e políticas, aspectos internacionais, tecnologia e pressão do tempo [Stair1999].

3.2.3 Visão Geral de Sistemas de Suporte à Decisão

O SSD é uma ferramenta que as empresas podem utilizar para dar apoio à tomada de decisões específicas de um problema. Devido ao seu conjunto organizado de pessoas, procedimentos, softwares, bancos de dados e outros dispositivos, os SSD oferecem uma maior eficácia na tomada de decisões de problemas semi-estruturados ou não estruturados. Eles devem ser projetados, desenvolvidos e aplicados para ajudar as organizações a atingirem suas metas e objetivos. Com isso, possibilitam a geração de maiores lucros, menores custos e melhores produtos e serviços [Stair2005].

Os SSD podem ser utilizados em todos os níveis gerenciais de uma organização e ainda podem apoiar tomadas de decisões gerenciais relacionadas com processos empresariais de valor adicionado. Assim, os SSD devem auxiliar àqueles que precisam tomar decisões em todos os níveis e em todos os aspectos do processo de tomada de decisão, considerando que são pessoas e não máquinas, que tomam as decisões [Stair2005].

A. Características e Recursos de um Sistema de Suporte à Decisão

Os SSD se apresentam como eficazes ferramentas de apoio gerencial devido as suas características de: (i) manipular grandes volumes de dados; (ii) obter e processar dados de fontes diferentes; (iii) proporcionar flexibilidade de relatórios e de apresentação; (iv) possuir orientação textual e gráfica; (v) executar análises e comparações complexas e sofisticadas, utilizando pacotes de softwares avançados; (vi) dar suporte às abordagens de otimização, satisfação e heurística; e (vii) executar análises por simulações e metas [Laudon2000].

Assim, antes mesmo de aplicá-los, deve-se avaliar também quais características serão utilizadas, visando à aplicação de uma ferramenta que seja ajustada para o problema em questão, e bem equilibrada quanto a custos, controle e complexidade [Laudon2000].

Os SSD devem ser mais flexíveis que os SIG, pois têm o potencial de auxiliarem os tomadores de decisão em diversas situações diferentes. Com isso precisam disponibilizar uma série de recursos. Entre eles destacam-se [Stair1999]:

- Suporte nas fases da solução de problemas;
- Suporte nas diferentes frequências de decisão;
- Suporte para diferentes estruturas de problemas;
- Suporte para diversos níveis na tomada de decisão.

Nota-se que as diversas características e os vários recursos podem não estar presentes em todos os SSD; a presença destes é determinada pela utilização e escopo do sistema.

3.3 Sistemas de Informação para Cana-de-Açúcar

A indústria canavieira dispõe de vários sistemas de informação que operacionalizam, organizam, apóiam e sustentam esta atividade econômica. Estes sistemas estão presentes nas várias etapas da atividade, do plantio até a distribuição final dos produtos.

Um dos maiores representantes destes sistemas é o PIMS® da empresa Próxima, ele possui módulos responsáveis por: planejamento e controle de atividades e recursos, visualização geográfica, gestão da manutenção industrial, pagamento de fornecedores e parceiros, entre outros [Proxima2008]. A iLab utiliza sistemas especialistas para apoiar nas decisões de plantio, colheita e no controle de frota [iLab2008]. Existem outros sistemas de outras empresas que também atuam neste mercado, mas citamos estes que são bastante representativos e utilizados nas usinas.

Além destes sistemas, existem várias pesquisas acadêmicas relacionadas com este setor. Alguns destes são: modelos de estimativa de produtividade de cana-de-açúcar utilizando dados agrônômicos, meteorológicos e de sensoriamento remoto [Rudorff1985] [Rudorff1995] [Fortes2003] [Picoli2006]; modelagem do sistema de produção [Keating1999]; análises espaçotemporais do NDVI (Índice de Vegetação da Diferença Normalizada) [Lucas2007]; previsões de safras a partir de análises temporais de índices de vegetação [Pontes2005]; RNA integrando dados espectrais e meteorológicos para estimativa de produtividade [Weber2005]; entre outros.

Outra linha de pesquisa vem se desenvolvendo desde 1997. Lima Neto propôs a utilização de RNA para realizar estimativa de indicadores agrônômicos da cana-de-açúcar, tais como: PCC, TCH e FIBRA [LimaNeto1997] [LimaNeto1998]. Outros trabalhos foram desenvolvidos com o objetivo de melhorar as predições destes indicadores [Pacheco2005] [Trigo2005]. Até aquele momento, os trabalhos apenas produziam novas informações para os tomadores de decisão, mas ainda não apoiavam nas decisões. O primeiro SSD desenvolvido utilizava as predições dos indicadores e as combinava em modelos matemáticos de decisão simples, que utilizava o método lexicográfico em suas sugestões de colheita [Pacheco2006]. Na seqüência, modelos de decisão

com algoritmos genéticos foram utilizados para melhorar as indicações de colheita [Oliveira2006]. Essa expertise acumulada ao longo dos últimos anos de pesquisa possibilitou também o desenvolvimento de um formalismo para suporte à decisão inteligente (HIDS), o qual será detalhado na seção seguinte.

Este trabalho de mestrado, por sua vez, é a evolução desses estudos. Ele sugere e avalia decisões de colheita de cana-de-açúcar com princípios multi-objetivos e o tratamento de aspectos logísticos. A pesquisa propõe ainda modificações em HIDS [LimaNeto2007], de modo que o novo formalismo (MO-HIDS) [Pacheco2008a] seja capaz de tratar problemas com abordagens realmente multi-objetivas (rever Figura 2.5) e não apenas com modelos agregadores de função (rever Figura 2.4).

3.3.1 Hybrid Intelligent Suite for Decision Support - HIDS

A suíte híbrida inteligente para suporte à decisão é a formalização do modelo de decisão proposto por Oliveira *et al.* [Oliveira2006] dentro de um sistema que integra técnicas preditivas e de buscas [LimaNeto2007].

A ontologia de decisão mencionada acima está ilustrada na Figura 3.4. Partindo de um espaço de decisões possíveis D , considera-se que cada decisão d_n é composta por uma ou mais componentes de decisão c_i . Cada componente, por sua vez, pode ser decomposta em atributos que a caracterizam. Baseados nessa ontologia foram definidos critérios de validade (*i.e.* restrições) que têm atuação sobre os atributos individuais de uma componente (*e.g.* não se pode cortar lotes com PCC abaixo de 14) ou sobre uma decisão como um todo (*e.g.* não se pode cortar uma extensão de lotes maior que mil hectares em um único dia de colheita).

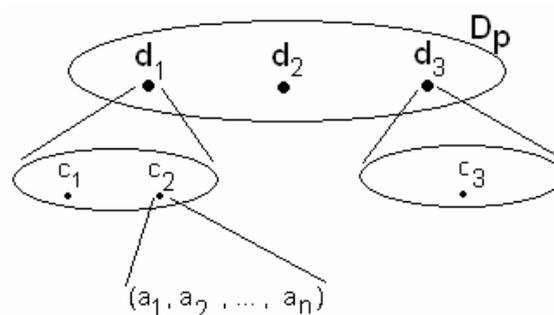


Figura 3.4. Ontologia de decisão proposta por Oliveira *et al.* [Oliveira2006]. Cada decisão pode ser decomposta como componentes e seus atributos (*i.e.* variáveis).

Descartadas as decisões inválidas, calcula-se a relevância de cada componente, de acordo com a equação (12). A relevância R considera pesos e funções objetivas definidas sobre cada um

dos atributos. Por fim, calcula-se a adequação de uma decisão F , a partir do somatório das relevâncias de suas componentes (13).

$$R(c_j) = \frac{\sum_{i=0}^n w_i * f_i(a_i)}{\sum_{i=0}^n w_i}, \quad (12)$$

$$F(d_k) = \sum_{j=0}^n R(c_j). \quad (13)$$

O HIDS (sigla do inglês *hybrid intelligent decision suite*) é uma proposta de implementação dos conceitos supracitados para a avaliação de decisões. A visão geral da sua arquitetura pode ser vista na Figura 3.5. Nela observa-se uma distinção clara entre fatores e indicadores: estes primeiros representam aspectos que possam causar impactos relevantes nas decisões; enquanto estes últimos representam maneiras de avaliar quão bons ou ruins são os resultados de uma decisão. O módulo preditivo é utilizado para gerar cenários futuros, explorando o espaço de decisão guiado pela técnica do módulo de busca.

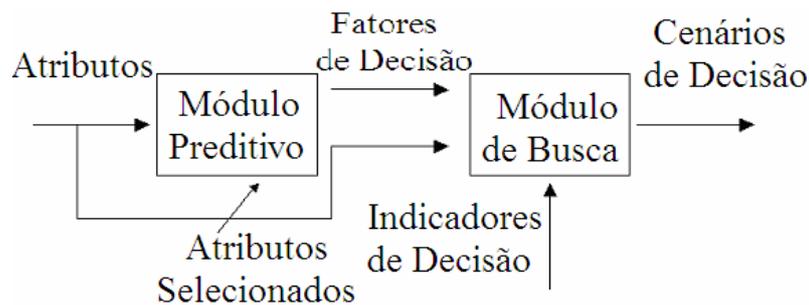


Figura 3.5. Visão geral do *hybrid intelligent decision suite*, adaptado de [LimaNeto2007].

Como benefício adicional de combinar as técnicas inteligentes contidas no módulo preditivo e de busca, a partir de diferentes configurações, é possível:

- Explorar o espaço de decisão para flexibilizar os indicadores de decisão a fim de reduzir ou aumentar certos critérios de validade para localizar soluções que seriam desprezadas em uma busca convencional [Oliveira2007];
- Contornar o problema inverso, ou seja, quando se conhece a decisão desejada, mas deseja-se descobrir como (quais) atributos geram esta decisão (*i.e.* mapeamento imagem-domínio) [LimaNeto2007].

Capítulo 4

Uma Abordagem Evolucionária Multi-Objetiva para Suporte à Decisão de Colheitas de Cana-de-Açúcar

Neste capítulo, descreve-se a contribuição teórica desta dissertação mestrado. Para tal, inicia-se com a formalização do problema solucionado, seguido por uma descrição geral do sistema proposto. Por se tratar de uma aplicação em um problema real e também para facilitar o entendimento da proposta, a contribuição foi desenvolvida e está apresentada em etapas, nas quais está detalhada a trajetória da contribuição.

4.1 Definição do Problema

A cultura de cana-de-açúcar, como descrita na seção 3.1, evolve uma série de fatores que devem ser considerados durante todo o seu desenvolvimento. A colheita da cana, em particular, tem importância direta na produtividade (rentabilidade) da atividade como um todo; além de requerer decisões complexas em tempo exíguo acerca de grandes volumes de dados.

A colheita de uma safra agrícola de lotes cultivados com cana-de-açúcar (em geral todos os lotes) é obtida como o conjunto das decisões de colheita (e sua execução) nos vários dias de trabalho durante a fase de colheita, ou seja, a integração das seleções diária Ω de um subconjunto de lotes que respeite as restrições de frente de corte, de transporte e de moagem.

Como em qualquer atividade econômica, as escolhas a serem realizadas devem favorecer dois princípios administrativos básicos: o aumento da produtividade e a redução dos custos. Neste cenário, a produtividade é fortemente determinada pela maximização dos índices de PCC,

FIBRA e ATR; e minimização dos custos de corte e transporte que são calculados a partir da distância do lote a ser colhido até a unidade de moagem. Além desses fatores, existem outros, ligados ao plantio, tratos culturais e industriais – todos não integrantes da colheita.

Em geral, como as áreas cultivadas são bastante extensas, com diversas variedades de cana, de solo e sob diferentes condições climáticas, as centenas de lotes l_i apresentam distintos retornos produtivos para um mesmo período de colheita. Este fato, combinado com as possíveis logísticas de corte adotada, transforma as tarefas associadas com a colheita num problema combinatorial multi-objetivo (NP-completo). Este fato demanda tecnologias computacionais poderosas para que se encontre soluções aceitáveis em tempo hábil.

Assim como nas usinas sucro-alcooleiras, o lote é a unidade de produção, também no sistema proposto para apoio a decisão de colheita devem ser identificados quais lotes que deverão ser colhidos durante cada período considerado. A Tabela 4.1 descreve algumas outras variáveis a serem consideradas no processo de colheita da cana-de-açúcar.

Tabela 4.1. Variáveis consideradas no problema da colheita da cana-de-açúcar.

Variáveis	Descrição	
Colheita	$\Omega = \{l_{23}, l_{900}, \dots, l_x\} \wedge \Omega \subset \Psi = \{l_1, \dots, l_n\} \mid x \leq n$	
Lote	<i>PCC</i>	
	<i>FIBRA</i>	Atributos qualitativos
	<i>ATR</i>	
	<i>TCH</i>	Atributo de produtividade
	<i>Área</i>	
	<i>Distância</i>	Atributos logísticos
	<i>Id</i>	
Dados industriais	<i>capacidade de moagem (CM)</i>	
	<i>tonelagem desejada (TD)</i>	

Durante o desenvolvimento deste projeto, a própria formulação do problema foi modificada. Notadamente, alguns trabalhos avaliavam dias isolados de colheita [Oliveira2007] [Pacheco2008b], enquanto que outros, abordavam a safra como um todo [Pacheco2007].

Assim, o problema da colheita de uma safra de cana-de-açúcar é formulado com quatro objetivos: (i) aumentar a produção de PCC (14), (ii) aumentar a produção de FIBRA (15), (iii) aumentar a produção de ATR (16) e (iv) reduzir os custos logísticos com transporte (distância percorrida) da cana-de-açúcar (17); as duas restrições impostas referem-se à tonelagem produzida diariamente (TP) que deve estar entre os limiares da tonelagem desejada (TD) (18). Isto garante um processo industrial contínuo e sem desperdício por matéria prima acumulada. Os detalhes sobre como é realizado o cálculo da distância percorrida serão apresentados nas seções seguintes.

Matematicamente, tem-se: a minimização de $f(\vec{l}) = (f_1(\vec{l}), f_2(\vec{l}), f_3(\vec{l}), f_4(\vec{l}))$, onde:

$\vec{l} = (l_1, l_1, \dots, l_n)^T$ representa o conjunto de n lotes da usina e

$$f_1(\vec{l}) = -\sum_{y=1}^w \sum_{i=1}^n l_i * PCC_i * TCH_i * AREA_i, \quad (14)$$

$$f_2(\vec{l}) = -\sum_{y=1}^w \sum_{i=1}^n l_i * FIBRA_i * TCH_i * AREA_i, \quad (15)$$

$$f_3(\vec{l}) = -\sum_{y=1}^w \sum_{i=1}^n l_i * ATR_i * TCH_i * AREA_i \quad e \quad (16)$$

$$f_4(\vec{l}) = -\sum_{y=1}^w \text{calculaDistancia}_y(\vec{l}) \quad (17)$$

são os objetivos da colheita. Estes, restritos à:

$$98\%TD \leq \sum_{i=1}^n l_i * TCH_i * AREA_i \leq 105\%TD, \quad (18)$$

onde: $\sum_{i=1}^n l_i * TCH_i * AREA_i = TP$, a tonelagem produzida diariamente.

Como mencionado anteriormente, em algumas fases do desenvolvimento deste trabalho o problema da colheita de cana-de-açúcar considerou apenas 1 (um) dia de colheita ao invés de toda safra. Assim, para estes casos, o número de períodos utilizados na colheita é fixado em um – $w = 1$.

4.2 Visão Geral do Sistema de Apoio à Decisão

Sistemas inteligentes híbridos (HIS, sigla do inglês *hybrid intelligent systems*) são bem sucedidos no tratamento de problemas que demandam a realização de tarefas computacionais distintas. Em particular, problemas de tomada de decisão são bons candidatos para aplicações de HIS por suas freqüentes naturezas duais. Isto porque o suporte à tomada de decisão geralmente envolve duas fases: (i) predição de cenários de decisão e (ii) buscas nestes cenários. O sistema proposto inclui tratamento para ambas as fases. Adicionalmente, para reduzir as inconveniências (incertezas) inerentes aos esforços de tomada de decisão (como formulado acima) é desejável algum tratamento do “Problema Inverso”, como proposto em *Hybrid Intelligent Decision Suite* (HIDS) e mencionado na seção 3.3.1 [LimaNeto2007].

A abordagem desenvolvida e apresentada nesta dissertação, chamada de MO-HIDS [Pacheco2008a], estende HIDS com a inclusão de um módulo capaz de tratar adequadamente tarefas multi-objetivas (MO). Isto foi possível pela inclusão de técnicas de Otimização Multi-

Objetivo Evolucionária (EMOO, do inglês *Evolutionary Multi-Objective Optimization*). Esse novo módulo contribui com a criação de um *Pareto front* para cada cenário de decisão estimado por uma Rede Neural Artificial (RNA) de Múltiplas Camadas de Perceptron (MLP, do inglês *Multilayer Perceptron*) – o módulo de preditivo, já presente em HIDS. Finalmente, a fim de melhorar a indispensável interação com os decisores, pode-se ou não utilizar lógica *fuzzy*.

A aplicação de EMOO na modelagem de problemas reais apresenta vantagens quando comparada com sistemas apenas baseados em preferências [Deb2001].

4.2.1 Modelo Abstrato para Tomada de Decisão Multi-Objetiva

Assume-se que o conjunto de decisões D é composto por decisões viáveis D_f (*feasible*) e decisões inviáveis D_u (*unfeasible*). Focando nos objetivos em lugar das componentes de decisão (do modelo original do HIDS), cada decisão d_i engloba um conjunto de funções objetivo o_i que são dependentes de uma ou mais variáveis de decisão; como visto na Figura 4.1 o novo modelo abstrato de decisão estende a idéia apresentada na Figura 3.4.

Conceitualmente nas teorias multi-objetivas, existem dois possíveis espaços de busca: (i) o espaço das variáveis de decisão e (ii) o espaço dos objetivos [Deb2001]. No espaço de decisão existem os limites para cada variável enquanto as restrições são impostas no espaço objetivo. Em particular, o não cumprimento dessas restrições descarta soluções (*i.e.* torná-as inviáveis). Isto é representado pela região D_u na Figura 4.1.

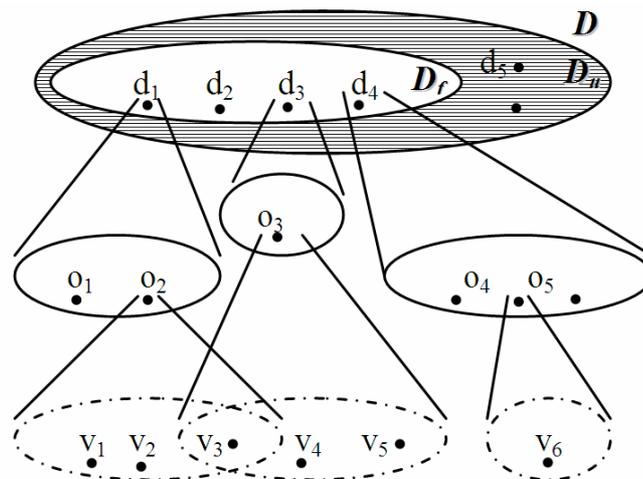


Figura 4.1. Nova ontologia de decisão que foca objetivos ao invés de componentes como no original HIDS [LimaNeto2007]. Cada solução viável pode ser decomposta como funções objetivos e suas variáveis de decisão.

Aparentemente pequenas, as implicações dessa nova abordagem para o novo modelo de decisão são bastante relevantes, especialmente, quando os componentes de decisão e os objetivos são comparados. Em HIDS, os componentes de uma decisão são calculados por uma equação única que agrega funções ponderadas (ver na equação (12), pág. 68), enquanto que na nova abordagem os objetivos são as próprias funções. Assim, quaisquer componentes de HIDS podem ser diretamente comparados, enquanto que os objetivos de MO-HIDS são incomparáveis (no sentido que todos os objetivos têm sua própria importância).

No HIDS canônico a avaliação de uma decisão é o somatório dos valores assumidos por cada um de seus componentes (ver na equação (13), pág. 68), desse modo, as decisões podem ser totalmente ordenadas e, conseqüentemente, selecionada a supostamente melhor. Com a nova abordagem, desde que objetivos não são comparáveis individualmente, a comparação (avaliação) deve ser realizada diretamente no nível de decisão, utilizando o conceito de dominância [Deb2001]. Desta maneira, um conjunto de ótimas soluções (decisões) pode ser encontrado, representando os *trade-offs* entre as funções. Ou seja, a abordagem proposta possibilita uma comparação justa entre os objetivos a serem otimizados.

4.2.2 Incluindo Habilidades Multi-Objetivos em HIDS

O impacto direto desta nova abordagem é a eliminação das equações de agregação (12) e total (13), em outras palavras, a exclusão dos pesos de ponderação w_i , que muitas vezes são difíceis de obter. Note que agora, o decisor terá que escolher uma solução dentre o conjunto de soluções ótimas encontrado (sem precisar julgar a relevância entre objetivos). Alguns podem entender isto como uma má contribuição, pois se cria um esforço adicional. Entretanto, Deb defendeu que o uso de informações preferenciais de alto nível depois da otimização (decisão *a posteriori*) traz mais benefícios que o contrário (decisão *a priori*) [Deb2001]. A explicação é simples: o uso das preferências após a otimização permite medir e comparar as melhores soluções sem interferência externa. Como bônus, diferentes cenários de decisão podem ser facilmente gerados, sem os custos adicionais de novas simulações.

Outra possível preocupação decorrente da abordagem adotada é o fato de que a escolha da melhor solução pode ser complexa ou pode consumir bastante tempo, a depender do tamanho do conjunto de soluções ótimas e da complexidade do problema (*i.e.* número de variáveis). Para auxiliar os decisores neste possível problema adicional, um terceiro módulo – heurístico – foi adicionado à proposta. Este módulo foi concebido com o objetivo de ajudar na seleção de uma solução única do Pareto *front* produzido pelo módulo MO de busca.

Os pesos descartados (ponderações entre os objetivos definidas pelo decisor), anteriormente utilizados para guiar as buscas do HIDS, retornam conceitualmente ao sistema MO-HIDS nesta busca *a posteriori*. Isso, para tornar o processo decisório mais real e intuitivo. Nesta nova interação entre decisor-sistema, provida pelo módulo heurístico, considera-se também preferências nominais em vez de apenas valores brutos (*crisp*); *i.e.* foi adicionado lógica *fuzzy* neste último componente. O módulo heurístico incorpora também a experiência dos tomadores de decisão e usa as métricas dependentes do problema para escolher a melhor decisão.

A Figura 4.2 apresenta a estrutura híbrida do MO-HIDS. Nesta abordagem o módulo preditivo é utilizado para estimar os indicadores agronômicos da cana-de-açúcar. O módulo de busca realiza a otimização gerando combinações diferentes de lotes para a colheita. O módulo heurístico, a partir das preferências do decisor, indica qual das combinações é a melhor para o problema. A ligação entre o módulo MO e o módulo preditivo deve existir, pois em alguns problemas, o próprio cálculo das funções objetivas é realizado a partir de previsões.

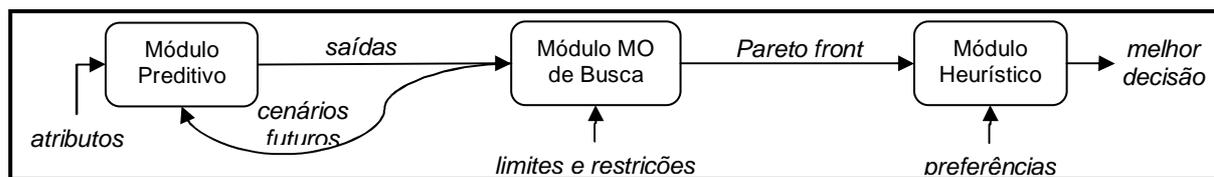


Figura 4.2. Visão geral do novo MO-HIDS com habilidades de suporte à decisão multi-objetivo.

4.3 Evolução da Proposta

Nesta seção são descritos como ocorreram os avanços desta pesquisa. Detalha-se o ponto inicial da pesquisa e são detalhadas todas as contribuições da pesquisa para o problema.

A presente pesquisa inicia-se como continuidade de trabalhos anteriores [Pacheco2006] [Oliveira2006] [Alencar2006]. Nesses trabalhos, as decisões de colheita eram bastante simples e não consideravam qualquer aspecto logístico ou a multiobjetividade do problema. Essencialmente, foram os primeiros módulos de decisão acoplados às proposições iniciais de Lima Neto de estimar indicadores agronômicos para colheita de cana-de-açúcar a partir de uma RNA [LimaNeto1998].

Ainda nessa linha de pensamento, mas já como parte inicial desta nova proposta, desenvolveu-se um modelo que utilizava um módulo decisor fuzzy para combinar os indicadores inferidos para sugestão de colheita [Oliveira2007]. A partir daí, há uma mudança importante no foco da pesquisa que passa a ser dividida em três fases:

- Formulação multi-objetiva confrontando algoritmos clássicos e evolucionários (MOEA) e avaliação considerando toda a safra [**Pacheco2007**];
- Instanciação do problema como um problema da mochila permitindo comparações entre representações, algoritmos e operadores; e objetivos logísticos passam a ser tratados no modelo [**Pacheco2008b**];
- Aplicação em dois níveis de MOEA revelando os impactos causados na produtividade da safra quando diferentes preferências (heurísticas) são aplicadas durante a colheita [**Pacheco2008c**].

4.3.1 Fase Um – Formulação Multi-Objetiva

Como mencionado anteriormente, os trabalhos anteriores de Pacheco [**Pacheco2006**], Oliveira [**Oliveira2006**][**Oliveira2007**] e Alencar [**Alencar2006**] trataram decisões de forma simples. Ou seja, eles consideravam apenas um dia de colheita a cada interação de decisão e não eram multi-objetivos. Nesses trabalhos, apesar da essência do problema ser multi-objetiva, os objetivos de decisão eram tratados individualmente ou de maneira conjunta (ponderada).

Em geral, os objetivos de trabalho desta fase foram:

- Definir uma formulação MO inicial para o problema da colheita da cana-de-açúcar⁵;
- Comparar desempenho entre algoritmos clássicos e evolucionários quando aplicados ao problema MO de colheita da cana-de-açúcar;
- Modificar a formulação do problema para realização da avaliação em toda a safra e não em dias isolados de colheita. Isto com vistas a maximizar a produção da safra e não apenas de uma parcela dessa.

A visão do sistema nessa fase já sinalizava para um esboço da proposição feita em MO-HIDS (descrita na seção 4.2). Ela era composta basicamente por dois módulos, um preditivo (RNA) e um otimizador (EMOO). A heurística de decisão aplicada a cada otimização era fixa, e estabelecia uma priorização na produção de PCC. Com o amadurecimento destas heurísticas, foi adicionado um terceiro módulo – o decisor [**Pacheco2008a**].

Duas formulações MO foram propostas aqui, diferentes apenas no modo de tratar as restrições do problema, a saber:

⁵ A base de dados utilizada na fase um desta pesquisa não disponibilizava dados com o indicador agrônomo ATR nem, tão pouco, dados logísticos.

- a) Formulação-1: Considera apenas dois objetivos, maximizar as produções de PCC e de FIBRA, equações (22) e (23), respectivamente. A restrição a cerca da tonelagem desejada é absorvida utilizando uma função de penalidade inserida no cálculo das respectivas funções *fitness*, que são modificações das equações (14) e (15), propostas na seção 4.1. O fator de penalidade f_{pen} (21) é calculado a partir da taxa na diferença entre a tonelagem produzida e a desejada (20). Como os prejuízos são maiores quando uma usina interrompe suas operações do que quando a matéria prima é desperdiçada (*i.e.* apodrece no pátio), uma penalidade maior foi atribuída ao primeiro caso.

$$dif = TP - TD \quad (19)$$

$$tx_{dif} = \frac{dif}{TD} \quad (20)$$

$$f_{pen} = \begin{cases} 20 \therefore tx_{dif} < -2\% \\ 10 \therefore tx_{dif} > 5\% \\ 0 \therefore -2\% \leq tx_{dif} \leq 5\% \end{cases} \quad (21)$$

$$f_1(\hat{l})_{pen} = f_1(\hat{l}) - f_{pen} * \left(\frac{f_1(\hat{l})}{TP} \right) * |dif| \quad (22)$$

$$f_2(\hat{l})_{pen} = f_2(\hat{l}) - f_{pen} * \left(\frac{f_2(\hat{l})}{TP} \right) * |dif| \quad (23)$$

- b) Formulação-2: Considera a restrição TD como um novo objetivo. A otimização é realizada considerando três objetivos: maximização na produção de PCC, maximização na produção de FIBRA, e minimização da diferença entre a TP e TD. As equações que descrevem estes objetivos são, respectivamente: (14), (15) e (24).

$$\psi = |dif| \quad (24)$$

O conjunto de algoritmos escolhido para desempenhar estas otimizações foi composto pelos representantes clássicos: método lexicográfico [Linkov2004], MAUT e AHP [Fülöp2005], e ELECTRE [Figueira2004]; e por versões evolucionários de primeira e segunda geração: VEGA [Schaffer1984] e NSGA-II [Deb2000a].

Finalmente, com a necessidade de uma avaliação mais prolongada nas sugestões da colheita, pela primeira vez as comparações finais entre os algoritmos utilizados consideravam o somatório de toda colheita. Note que nesta fase não foram considerados a produção de ATR e a redução dos custos logísticos, pois a base de dados utilizada nos experimentos não possui tais atributos.

4.3.2 Fase Dois – Incorporação de Aspectos Logísticos e Paradigma do Problema da Mochila

Após a comprovação de que o uso de MOEAs auxiliam os tomadores de decisão, bem como possibilitam uma melhora na produtividade da safra [Pacheco2007], como descrito na *Fase Um*, inicia-se a fase seguinte. O trabalho de investigação prosseguiu no sentido de dotar a solução proposta de mais conexão com as especificidades do problema, ou seja, incorporar aspectos logísticos no modelo de decisão.

De imediato, foram identificados similaridades no processo da colheita da cana-de-açúcar com o tradicional problema da mochila, mais ainda, com sua versão MO – o problema das múltiplas mochilas [Pisinger1995].

Verificou-se também, que a contribuição gerada durante a fase anterior, apesar de bastante relevante, poderia nunca vir ser usada na prática já que suas sugestões de colheita tinham preocupações exclusivamente agrônômicas. Diante deste novo cenário e com as aquisições de duas novas bases de dados⁶, iniciou-se esta fase com os seguintes objetivos:

- Reformular o problema da colheita da cana-de-açúcar utilizando o paradigma das múltiplas mochilas. Especificamente, refletir sobre novas formas de representação dos dados, algoritmos e operadores genéticos.
- Incorporar dados logísticos à formulação do problema, tornando-o viável sua aplicação em ambientes reais de colheita.

Nesta fase, com o foco voltado também para a definição MO do problema, todas as avaliações consideraram apenas um dia da colheita, já que a avaliação de toda uma safra é o somatório de todos os dias de colheita.

A. O Problema da Colheita Visto Como um Problema de Múltiplas Mochilas

Resolver um problema específico pode demandar passos adicionais com relação à incorporação de algumas especificidades do problema dentro de modelos mais gerais. Isto pode ser visto como uma abordagem *bottom-up*. Entretanto, para ser mais efetivo na resolução de problemas reais, pode-se tentar identificar a classe do problema (*i.e.* uma abordagem *top-down*) e com isso acelerar a velocidade da obtenção de resultados pela utilização de modelos previamente concebidos para tratar da classe do problema em mãos.

⁶ As bases de dados utilizadas são compostas por dados reais e foram cedidas por duas usinas do interior do estado de São Paulo. De acordo com as políticas acordadas de proteção de informação, seus nomes não serão revelados.

Neste sentido, durante a evolução desta proposta, percebeu-se as similaridades do problema em lide com o clássico problema da mochila 0/1, em particular, o problema das mochilas múltiplas 0/1 (MKP, sigla do inglês *multiple 0/1 knapsack problem*). Utilizando este paradigma, a reformulação do problema da colheita da cana-de-açúcar passa a incorporar as seguintes assunções:

- Mochilas (*knapsacks*) – são quatro, cada uma representada por um dos objetivos do problema, a saber: três objetivos relacionadas a indicadores de agrônômicos ($f_1(\hat{l})$, $f_2(\hat{l})$ e $f_3(\hat{l})$) e uma novo novo objetivo relacionado a logística ($f_4(\hat{l})$) – respectivamente, equações (14), (15), (16) e (17);
- Objetos – são os lotes cultivados com cana-de-açúcar (l_i). Cada lote apresenta seus próprios atributos (descritos na Tabela 4.1) que indicam qualidade, produtividade e logística. Uma vez selecionado, o objeto (*i.e.* lote) pertencerá a todas as mochilas;
- Valor dos objetos (*profit*) – para cada uma das mochilas a valoração é representada, respectivamente, pelas quantidades de PCC (25), FIBRA (26) e ATR (27); e a distância do lote à usina;

$$q_i^1 = PCC_i * TCH_i * AREA_i, \quad (25)$$

$$q_i^2 = FIBRA_i * TCH_i * AREA_i, \quad (26)$$

$$q_i^3 = ATR_i * TCH_i * AREA_i. \quad (27)$$

- Capacidade da mochila – é determinada pela tonelagem desejada (TD) pela usina, ou seja, a mesma capacidade é compartilhada por todas as mochilas. A TD fornecida por um gerente agrícola reflete as necessidades industriais e capacidade de moagem respeitando as limitações das frentes de corte. Em outras palavras, a TD objetiva um contínuo fornecimento para as demandas de moagem durante toda colheita. Como esse valor não é alcançado precisamente (*i.e.* é improvável que isso ocorra), em termos multi-objetivos, a TD é transformada em duas restrições com as flexibilizações aceitas para mais ou para menos quando do fornecimento.

Robert Hinterding [Hinterding1999] definiu diferentes formas de representação e tratamento de restrições para o problema da mochila. Christine Mumford [Mumford2003] as adaptou para a versão MO e testou-as usando um *benchmark* de problemas MO proposto por Zitzler e Thiele [Zitzler1999b]. Na seqüência, novos MOEA também foram submetidos a estes problemas [Colombo2005]. Tudo isto serviu de inspiração para aplicar diferentes representações, operadores e algoritmos no problema da colheita da cana-de-açúcar.

Como o problema pode ser representado de várias maneiras, torna-se necessário utilizar operadores específicos para lidar com as particularidades de cada representação. Assim, para realizar a avaliação desta fase foram criadas quatro abordagens, *i.e.* quatro configurações de operadores combinados ao modo de representação dos dados (ver Tabela 4.2):

Tabela 4.2. Configurações de representação, tratamento de restrição e operadores genéticos adotados do problema da mochila e aplicadas no problema da colheita.

Identificador	Representação	Decodificador	Cruzamento	Mutação
bit	bit string	-	um ponto	bit flip
bitD	bit string	<i>best fit</i>	um ponto	bit flip
CX	baseada na ordem	<i>first fit</i>	ciclo	Swap
PMX	baseada na ordem	<i>first fit</i>	<i>partially matched</i>	Swap

- A primeira abordagem (*bit*) usa a representação binária sem qualquer função de penalidade ou decodificador. Ela trata puramente do processo evolucionário usando os operadores de um ponto (cruzamento) e de bit flip (mutação). É responsabilidade do algoritmo satisfazer as restrições do problema;
- A segunda abordagem (*bitD*) também usa a representação binária e os operadores da abordagem anterior. Entretanto, incorpora o uso do decodificador *Best Fit*⁷ para resolver as restrições, garantindo que apenas mochilas válidas fossem produzidas. Michalewicz e Arabas demonstraram, em problemas da mochila, que não existem vantagens em sobrescrever os cromossomos reparados na população [Michalewicz1994]. Mumford fez o mesmo para as versões MO do problema [Mumford2003]. Logo, nenhum dos experimentos deste trabalho utilizaram substituição dos indivíduos reparados;
- A terceira configuração (*CX*) usa a representação simbólica baseada na ordenação, o que requer o uso de decodificadores para produção de mochilas válidas já que todos os objetos (lotes) são apresentados em uma lista de permutação. Assim, esta abordagem combina a ordem, a heurística *First Fit*⁸, o operador de cruzamento de ciclo (*CX*) [Oliver1987] e o operador de mutação *swap*. *CX* foi anteriormente aplicado com sucesso em problemas de múltiplas mochilas [Mumford2003];
- Finalmente a última abordagem (*PMX*), que é bem semelhante à anterior, mas investiga como operadores de cruzamento diferentes se comportam no mesmo problema.

⁷ Não adiciona direto o primeiro item válido encontrado na mochila. Busca o maior (em termos de peso) válido dentre os demais para, então, adicionar.

O operador de cruzamento *partially matched* (PMX) [Goldberg1985] foi escolhido como sugerido em [Mumford2003].

Estas abordagens foram aplicadas em três dos MOEA considerados estado da arte: NSGA-II [Deb2000a], SPEA2 [Zitzler2001b] e PESA-II [Corne2001]. Espera-se a convergência de todos os algoritmos, no entanto, o uso destes combinados com diferentes representações para o problema específico da colheita da cana-de-açúcar pode revelar diferenças nos resultados finais. Desde modo, configuraram-se doze maneiras distintas para identificar uma que melhor solucione este complexo problema. Simulações destas distintas combinações estão detalhadas no capítulo seguinte desta Dissertação.

B. Incorporando Dados Logísticos no Problema da Colheita da Cana-de-Açúcar

Ao agregar logística na formulação do problema da colheita de cana-de-açúcar, aumenta-se a sua complexidade em relação à proposição anterior [Pacheco2007], ao passo que se aumenta o realismo e aplicabilidade do modelo de decisão concebido. Embora o ideal fosse dispor de toda distribuição dos lotes nos campos (*i.e.* coordenadas, distâncias entre eles, suas vizinhanças, etc), apenas a distância dos lotes para a usina estava disponível nas bases de dados utilizadas.

Para superar esta ausência parcial de dados, assumiu-se uma distribuição física hipotética dos lotes ao redor da usina baseada nas distâncias dos lotes à fábrica (ver Figura 4.3). Na figura, cada quadrado representa um lote (com áreas distintas) e vários possíveis tipos de vizinhança são demonstrados. Enquanto que um quadrado preto indica um lote selecionado para colheita, os cinza mais próximos são considerados seus vizinhos. O conceito de vizinhança de um lote é de suma importância, desde que reflète nas decisões de configuração de mão-de-obra, maquinário e transporte. Apesar de importantes, esses outros aspectos logísticos foram abstraídos neste modelo proposto.

Neste modelo o conceito de vizinhança é a definição do raio de atuação de uma frente de corte. Define os limites onde lotes podem ser cortados sem a necessidade de transporte mecânico. Para a colheita ser realizada, toda uma força de trabalho de uma frente de corte (*i.e.* mão-de-obra e máquinas para o corte e caminhões para transporte da produção) deve ser posicionada no campo para provimento da unidade de moagem. Se os lotes selecionados para a colheita estiverem distantes entre si (de acordo com o conceito de vizinhança) as forças de trabalho terão de ser deslocadas diversas vezes, aumentando assim os custos logísticos. Por outro lado, se os

⁸ Percorre a lista de elementos selecionados, considerando a ordem deles, e adiciona-os na mochila enquanto sua capacidade não é violada.

lotes selecionados estão dentro do raio de atuação desta frente, esta será movida apenas uma vez, consequentemente, reduzindo os custos da operação.

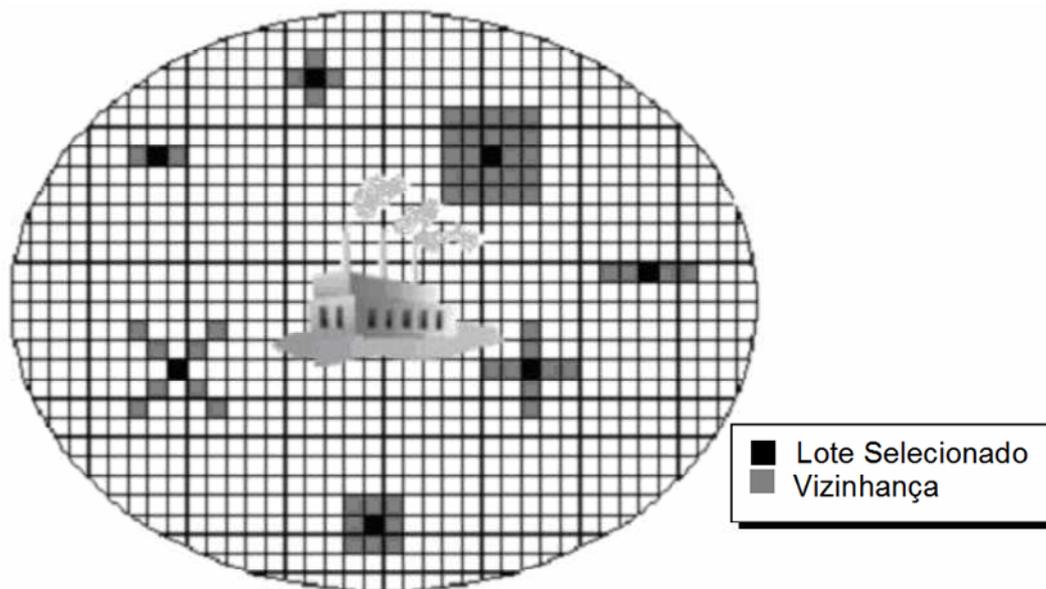


Figura 4.3. Distribuição física hipotética dos lotes de cana-de-açúcar e possíveis definições de vizinhança.

A heurística utilizada para calcular os custos de um dia de colheita considerou a distância total percorrida pelas frentes de corte de acordo com os seguintes passos:

- Agrupe todos os lotes selecionados para a colheita em vizinhanças;
- Para cada grupo, considere a distância percorrida como a média das distâncias de todos os lotes cortados;
- A distância total percorrida é o somatório das distâncias de todos os grupos.

Apesar desta assumpção hipotética relacionada aos posicionamentos dos lotes, isto em nada compromete o modelo proposto, haja vista que a avaliação é feita sobre o conceito de vizinhança utilizado e não sobre as reais posições dos lotes. Em ambientes industriais, onde certamente existe um mapeamento físico dos lotes, o modelo proposto continuará sendo válido e tão eficaz quanto demonstrado pelos experimentos deste projeto. Todos os experimentos consideraram o raio de atuação (critério de vizinhança) como oito vizinhos, visto na parte inferior da Figura 4.3.

4.3.3 Fase Três – Meta Aplicação de EMOO

Finalmente, com a formulação definida considerando aspectos agrônômicos e logísticos, e com a definição da melhor forma de representar o problema, bem como com a escolha do conjunto

técnico de algoritmos e operadores (vide *Fase Dois*) [Pacheco2008b], pode-se continuar a investigação por fatores ainda mais específicos do problema. Desta maneira, a terceira fase compreendeu buscar melhorias no diálogo entre os usuários (gerentes) e o sistema de suporte à decisão proposto.

As interações entre o usuário tomador de decisão e o sistema, basicamente, ocorrem nas parametrizações iniciais (por exemplo: tonelagem desejada, definição de vizinhança, etc), mas principalmente no módulo heurístico (*i.e.* decisor). Neste último momento é imprescindível os direcionamentos diários fornecidos pelo usuário no sentido de apontar as suas preferências. É importante ressaltar que por realizar uma otimização multi-objetivo, o segundo módulo retorna um conjunto com as soluções (decisões) ótimas – aquelas que conseguem ser representativas para todos os objetivos considerados (rever Figura 4.2). Restando assim, ainda, a incumbência da escolha da solução definitiva a ser implementada que pode ser realizada pelo (i) sistema (previamente configurado), (ii) pelo decisor ou (iii) por ambos, iterativamente.

De acordo com a definição do problema, a avaliação de produtividade de uma safra é medida a partir da soma de todos os dias de colheita (maiores detalhes na seção 4.1 e nas equações (14), (15), (16) e (17)). Deste modo, os objetivos desta fase são:

- Investigar os impactos causados na produtividade final de uma safra a partir das heurísticas (decisões) aplicadas nos dias isolados de colheita;
- Caso essas preferências se mostrem relevantes para o problema, desenvolver uma ferramenta que auxilie os tomadores de decisão nesta atividade.

Na *Fase um* [Pacheco2007], apenas um tipo de heurística foi utilizada em todos os dias de colheita e via de regra as maiores produções de PCC foram sempre consideradas como a preferência indicativa para as melhores decisões. Caso a preferência mudasse para FIBRA, por exemplo, as maiores produções de fibra seriam escolhidas em todos os dias da safra. Isso deveria indicar que se priorizarmos determinado objetivo e procedermos toda colheita com este mesmo foco, ao final, deveríamos alcançar a melhor produção possível para aquele objetivo. Intuitivamente a conclusão seria sim, no entanto após a aplicação de algumas seqüências de heurísticas aleatórias se observou que esta verdade não é absoluta. Este fato foi suficiente para satisfazer o primeiro objetivo apenas. Ou seja, a relevância das heurísticas aplicadas diariamente estava comprovada.

Para tratar adequadamente o segundo objetivo, concebemos mais uma aplicação de EMOO (meta-aplicação). Esta nova aplicação se compõe de dois níveis, no primeiro, define-se que heurísticas serão aplicadas nas seqüências de colheita (*i.e.* módulo combinador de

heurísticas). Já o segundo nível, é responsável por realizar a otimização da colheita em si, ou seja, é repetidamente aplicado para resolver as colheitas diárias – o problema da colheita de cana-de-açúcar propriamente dito. A Figura 4.4 ilustra o esquema adotado.

A. Problema do Primeiro Nível de Aplicação

Como o retorno (*i.e.* saída) da aplicação de segundo nível é um conjunto de aproximação e não uma solução única se faz necessário um processo de tomada de decisão. Para apoiar nesta seleção um MOEA é aplicado para prover a uma seqüência de heurísticas de colheita a partir de um conjunto de heurísticas previamente definido; a criação deste conjunto será detalhada a seguir. Este nível de aplicação determina ‘como’ as decisões de colheita serão tomadas.

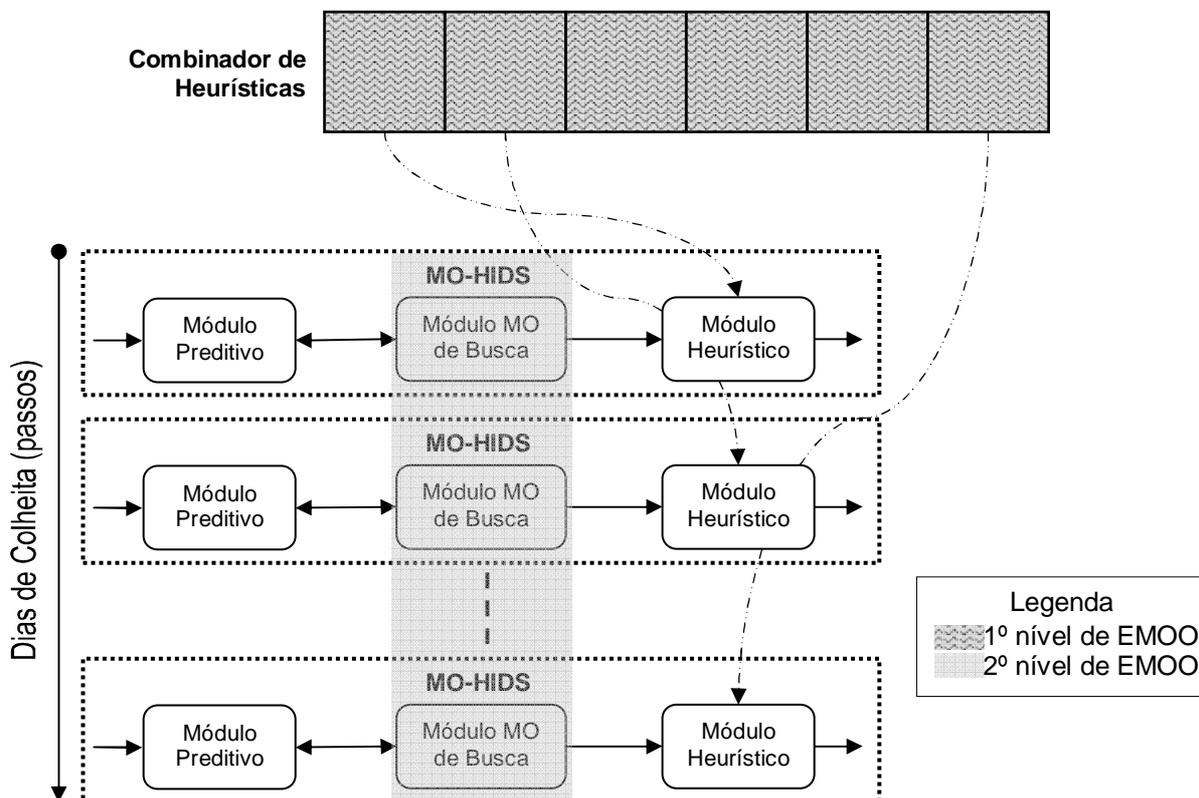


Figura 4.4. Sistema de colheita que usa uma meta-aplicação de EMOO para definir as heurísticas que serão utilizadas pelo outro nível de aplicação.

B. Problema do Segundo Nível de Aplicação

Este nível está localizado dentro do MO-HIDS [Pacheco2008a] (Figura 4.2), onde o problema da colheita da cana-de-açúcar é, de fato, solucionado. A tarefa desta otimização pode ser definida como encontrar um conjunto de lotes para serem colhidos que maximizem os objetivos agrônômicos – PCC, FIBRA e ATR; e ao mesmo tempo minimize o objetivo logístico

considerado – a distância total percorrida entre os lotes colhidos e a unidade de moagem. Os detalhes desta definição foram demonstrados anteriormente na seção 4.3.1. Depois desta otimização, a heurística definida pela aplicação do primeiro nível é aplicada, retornando ‘qual’ é a solução mais indicada dentre as candidatas ótimas.

C. Criação do Conjunto de Heurísticas

Com a proposta de uma seleção de preferências diária automática auxiliada por um MOEA, faz-se necessário a criação de um repositório de heurísticas que possibilitem tais escolhas. Para manter a compatibilidade entre todas as fases deste desenvolvimento foram criadas heurísticas que contém lógica *fuzzy* e outras puramente “aristotélicas” – clássicas. Assim, para o problema em questão foram criadas onze heurísticas, cinco normais (*i.e. crisp*) e seis *fuzzy*.

Quatro das heurísticas não-*fuzzy* representam diretamente os objetivos, já que retornam o valor máximo para seus respectivos fins. São elas *REDUZIR_CUSTO*, *AUMENTAR_PCC*, *AUMENTAR_ATR* e *AUMENTAR_FIBRA*. A quinta representante deste grupo tenta equilibrar as produções com os custos, *i.e.* retorna a solução com maior produtividade. Esta quinta heurística (Φ) é calculada a partir do somatório das produções de PCC, FIBRA e ATR dividido pela distância total percorrida (28).

$$\Phi = \left(\frac{\rho + \tau + \alpha}{\delta} \right) \quad (28)$$

As heurísticas *fuzzy* são definidas através de atribuições de três níveis de prioridade ou equivalências entre os objetivos. A cada otimização os objetivos de todas as soluções são normalizados de acordo com a variação relativa dentro do conjunto de aproximação retornado pela aplicação do segundo nível (*i.e.* Pareto *front*). Ou seja, a solução que apresentar o pior valor para um determinado objetivo recebe um número de classificação zero (neste objetivo) e a solução que contiver o melhor valor neste objetivo recebe a classificação um. As demais soluções são classificadas dentro deste intervalo, de acordo com a variação relativa de cada objetivo. Funções de pertinência classificam cada objetivo da solução em três classes: *RUIM*, *REGULAR* e *BOM* (veja a Figura 4.5). A maior prioridade aceita apenas as soluções que possuem objetivos classificados como *BOM*; a prioridade seguinte permite a seleção de soluções com objetivos *REGULAR* e *BOM*; por fim, a última prioridade admite qualquer solução.

Para uma solução ser elegível por uma heurística fuzzy ela tem que satisfazer a todos os critérios (*i.e.* prioridades) definidos para todos os objetivos. Se houver mais de uma solução elegível ou nenhuma, será escolhida a que apresentar o melhor valor no objetivo mais prioritário.

Se os critérios não forem priorizados e existirem mais de uma solução elegível (ou nenhuma), escolhe-se aleatoriamente uma entre as candidatas.

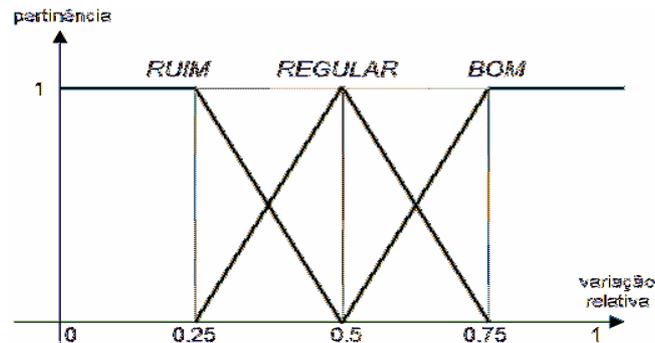


Figura 4.5. Funções de pertinência aplicadas na utilização das heurísticas fuzzy.

Com isso, as seis heurísticas fuzzy estão definidas nas linhas da Tabela 4.3 e são divididas em três categorias:

Tabela 4.3. Heurísticas fuzzy utilizadas e suas prioridades.

Categoria	Id	Prioridade		
		1 ^a	2 ^a	3 ^a
A	1	PCC	CUSTOS	FIBRA = ATR
	2	FIBRA	CUSTOS	PCC = ATR
	3	ATR	CUSTOS	PCC = FIBRA
B	1	PCC = FIBRA = ATR		CUSTOS
	2	CUSTOS		PCC = FIBRA = ATR
C	1	PCC = FIBRA = ATR = CUSTOS		

- Heurística fuzzy A: Prioriza um indicador agrônômico (IndA), depois os custos logísticos e finalmente os demais IndAs (*i.e.* heurísticas A1, A2 e A3 da Tabela 4.3);
- Heurística fuzzy B: Prioriza IndAs ou os custos logísticos (*i.e.* heurísticas B1 e B2 da Tabela 4.3);
- Heurística fuzzy C: Não faz priorizações, os objetivos são equivalentes (*i.e.* última linha da Tabela 4.3, C1).

A Tabela 4.4 exemplifica algumas soluções com a classificação de seus objetivos e suas elegibilidades dentro das seis heurísticas definidas acima.

Tabela 4.4. Seis exemplos de soluções e suas elegibilidades dentre as heurísticas fuzzy.

Solução	Objetivos				Elegível para
	PCC	FIBRA	ATR	CUSTOS	
1	<i>BOM</i>	<i>BOM</i>	<i>BOM</i>	<i>BOM</i>	todas
2	<i>REGULAR</i>	<i>BOM</i>	<i>RUIM</i>	<i>REGULAR</i>	A2 e B2
3	<i>RUIM</i>	<i>REGULAR</i>	<i>BOM</i>	<i>RUIM</i>	nenhuma
4	<i>BOM</i>	<i>REGULAR</i>	<i>REGULAR</i>	<i>BOM</i>	A1, B1, B2 e C1
5	<i>REGULAR</i>	<i>RUIM</i>	<i>REGULAR</i>	<i>REGULAR</i>	B2
6	<i>RUIM</i>	<i>RUIM</i>	<i>RUIM</i>	<i>RUIM</i>	nenhuma

Capítulo 5

Experimentos Realizados

Neste capítulo estão descritos os experimentos realizados e os resultados obtidos que confirmam as proposições feitas no capítulo anterior. Para manter simetria com a forma como a contribuição foi explicada, *i.e.* dividida em fases, os resultados também estão divididos desta maneira. Assim, o leitor pode mais facilmente perceber como a evolução da proposta gerou resultados cada vez melhores. Como os objetivos e a metodologia já foram descritos anteriormente, aqui estão descritas as bases de dados usadas, a modelagem neural do módulo preditivo, as parametrizações dos algoritmos evolucionários empregados e as métricas utilizadas em cada fase deste trabalho.

Por se tratar de um domínio de aplicação em problema real – a colheita da cana-de-açúcar, naturalmente demandou que sua validação ocorresse via a utilização de dados reais de produção. Para tanto, obteve-se e foram utilizadas três bases de dados reais que foram aplicadas em mais de uma fase. Para evitar repetição, essas bases estão descritas separadamente. Outros detalhes específicos, como a modelagem neural, também estão atrelados às bases de dados e, portanto, também antecederão as fases de evolução da proposta.

Apesar do foco principal da contribuição deste trabalho ser na otimização das decisões de colheita (vide o módulo otimizador MO), também o módulo preditivo não deve ser menosprezado já que ele é quem gera os dados necessários para a otimização. Destacamos portanto, o papel deste módulo na qualidade final dos resultados aqui apresentados.

5.1 Bases de Dados

Todas as três distintas bases de dados utilizadas em todas as fases desta pesquisa, assim como nos trabalhos anteriores citados várias vezes ao longo desta dissertação são compostas de dados reais

de três diferentes usinas do sudeste brasileiro. Como estes dados foram extraídos diretamente dos sistemas transacionais dessas indústrias, na forma disponibilizada, eles ainda não estavam preparados para serem utilizados pelas técnicas inteligentes aqui utilizadas.

É importante salientar que a maior parte dos dados de entrada é apresentada diretamente às RNA. Já os dados tratados evolucionariamente, em sua maioria, são resultados das predições (*i.e.* as saídas) dessas RNA (ver Figura 4.2). Por isso, foi mandatório a realização de um pré-processamento nestes dados a fim de que as diferentes escalas nos parâmetros quantitativos pudessem ser igualmente tratadas (*i.e.* utilizadas pelas RNA sem efeito de indução); já os dados qualitativos também foram rotulados para poderem ser representados adequadamente pelo sistema.

Para facilitar as referências ao longo do texto às três bases de dados utilizadas neste trabalho, adotou-se a seguinte denominação: CLIENTE_0, CLIENTE_1 e CLIENTE_2.

5.1.1 Base CLIENTE_0

Esta base de dados é a mais antiga de todas. Ela foi criada em 1997 para o trabalho de Lima Neto [LimaNeto1998] e foi utilizada em todos os trabalhos anteriores afins a este mestrado: [Pacheco2005], [Pacheco2006], [Alencar2006], [Oliveira2006] e [Oliveira2007]. Esta foi também a base de dados utilizada na *Fase Um* deste trabalho [Pacheco2007].

Por ser esta uma base de dados onde originalmente já havia sido realizado um pré-processamento, não foram necessárias outras manipulações. Esta base de dados contém histórico de três safras, distribuídos em 1671 padrões os quais foram separados em 50% para treinamento, 25% para validação e 25% para testes. Cada padrão (linha) representa um lote. E nas 13 colunas (*i.e.* campos) estão representados 10 atributos como segue. O algarismo ao lado de cada campo indica a ordem do campo em cada padrão:

Informações de entrada:

- a) Safra – 1;
- b) Variedade da cana-de-açúcar (escolhida a mais usual);
- c) Estágio da maturação (número do corte) – 2;
- d) Idade do corte (em dias) – 3;
- e) Data do plantio (aaaa/mm/dd) – 4, 5, 6;
- f) Data do corte (aaaa/mm/dd) – 7, 8, 9;
- g) Tipo do solo – 10.

Informações de saída:

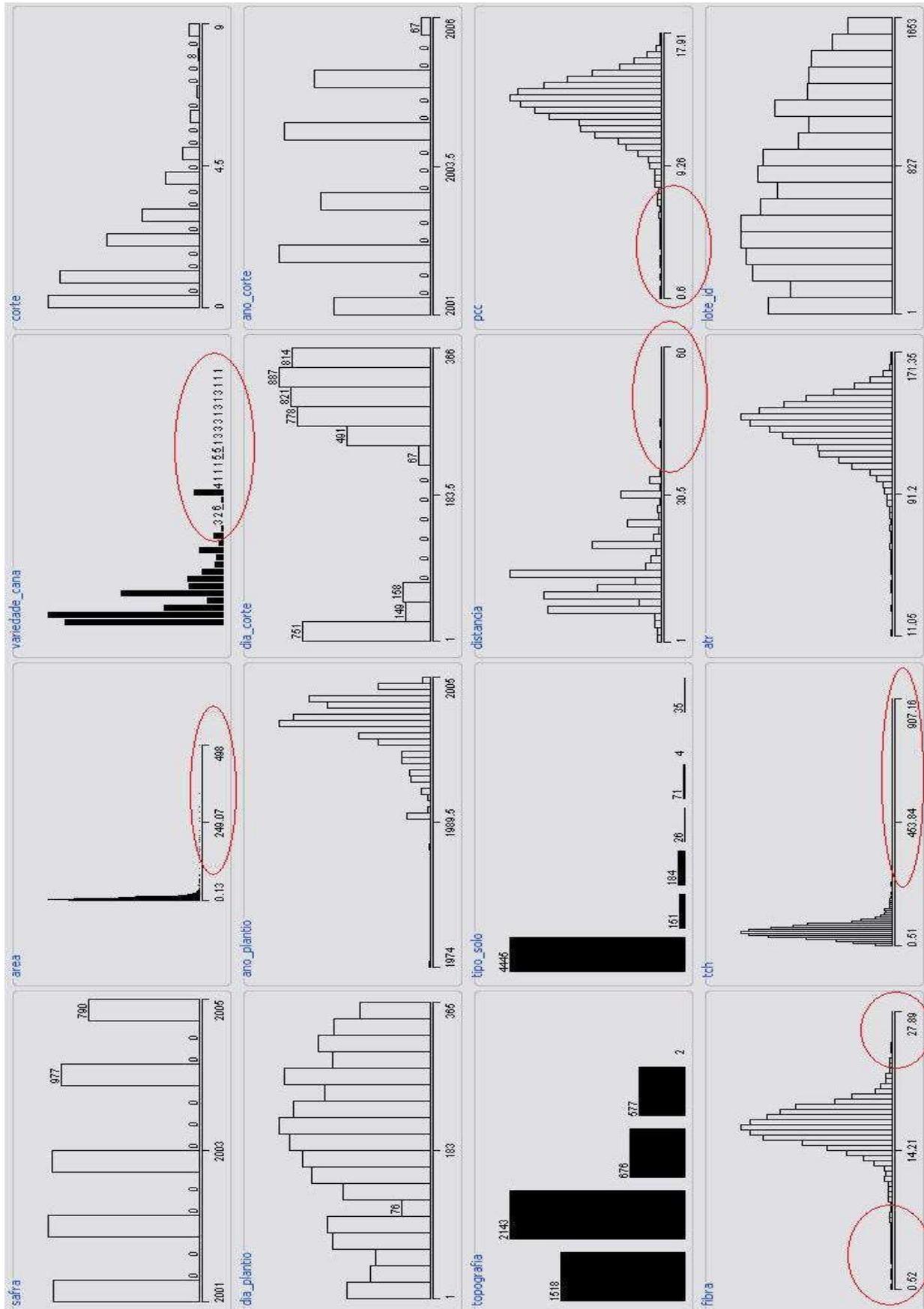


Figura 5.1. Base de dados CLIENTE_1 antes do pré-processamento: atributos e distribuição.

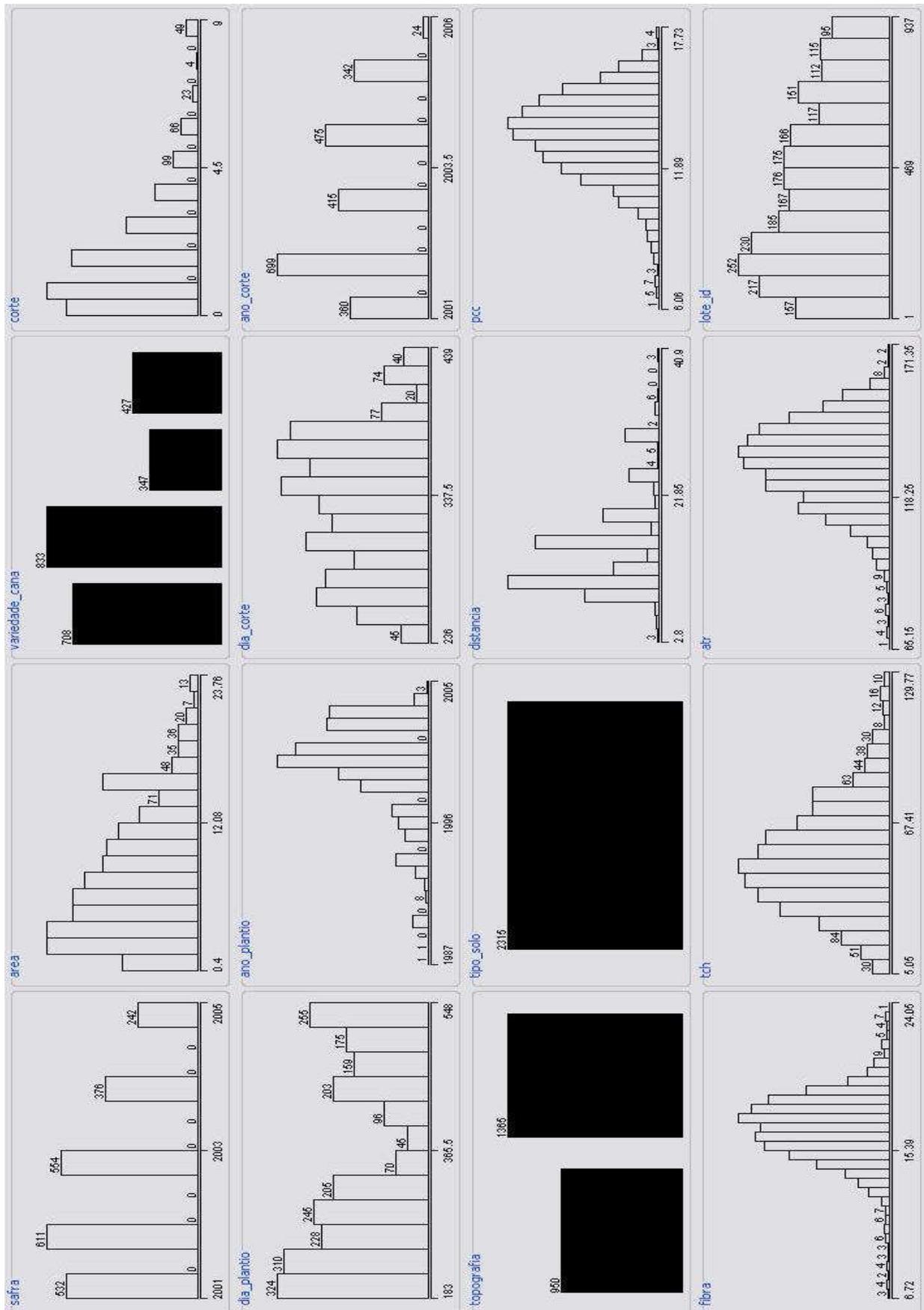


Figura 5.2. Base de dados CLIENTE_1 após o pré-processamento.

Observa-se ainda que os atributos variedade da cana, topografia e tipo de solo apresentam significantes desbalanceamentos em suas amostras. A permanência destes padrões na base de dados aumentaria a dificuldade para convergência da RNA utilizada para inferências. Assim, decidiu-se por remover dos padrões com esses atributos julgados também não muito relevantes dado o escopo da decisão a ser modelada. A Figura 5.2 ilustra a base de dados utilizada. Após todas as validações e remoções restaram 2315 padrões referentes a 937 lotes dispostos em um período total de 5 safras.

Uma safra de cana-de-açúcar é determinada pelo seu período de colheita que usualmente inicia-se em setembro e finaliza-se em março do ano seguinte. Para melhor ajustar a distribuição dos dias de colheita e plantio, desloca-se seis meses nessas datas. Na Figura 5.2 a distribuição do dia de colheita é normal, diferentemente do ilustrado na Figura 5.1 que mostra uma gaussiana invertida (*i.e.* muitos valores zerados no centro do gráfico). Experimentalmente, verificou-se que essa simples alteração implicou em melhores resultados de predição pelas RNA.

Tabela 5.2. Limites inferiores e superiores de atributos nas bases CLIENTE_1 e CLIENTE_2.

Atributo	CLIENTE_1		CLIENTE_2	
	<i>min</i>	<i>max</i>	<i>min</i>	<i>max</i>
Área	0	24	0	24
PCC	6	18	6	18
FIBRA	6	24.5	6	21.5
TCH	5	130	5	111
ATR	50	175	0	1.32

5.1.3 Base CLIENTE_2

Esta é uma base de dados menor – apenas dois arquivos – quando comparada com a base anterior, pois se refere apenas a duas safras (2002 e 2004). No entanto, qualitativamente ela é tão completa quanto a CLIENTE_1 em seus atributos. De modo que o procedimento utilizado em ambas as bases foi bastante semelhante.

Iniciou-se com uma pré-análise para remoção de padrões com atributos faltantes ou datas inválidas; na seqüência verificou-se os valores extremos e os padrões com atributos sem representatividade (limites definidos na Tabela 5.2). Esta base inicialmente possuía 1260 padrões que, após o processamento, resultou em apenas 681 padrões referentes a 590 lotes. A Figura 5.3 e a Figura 5.4 ilustram estes dois distintos momentos.

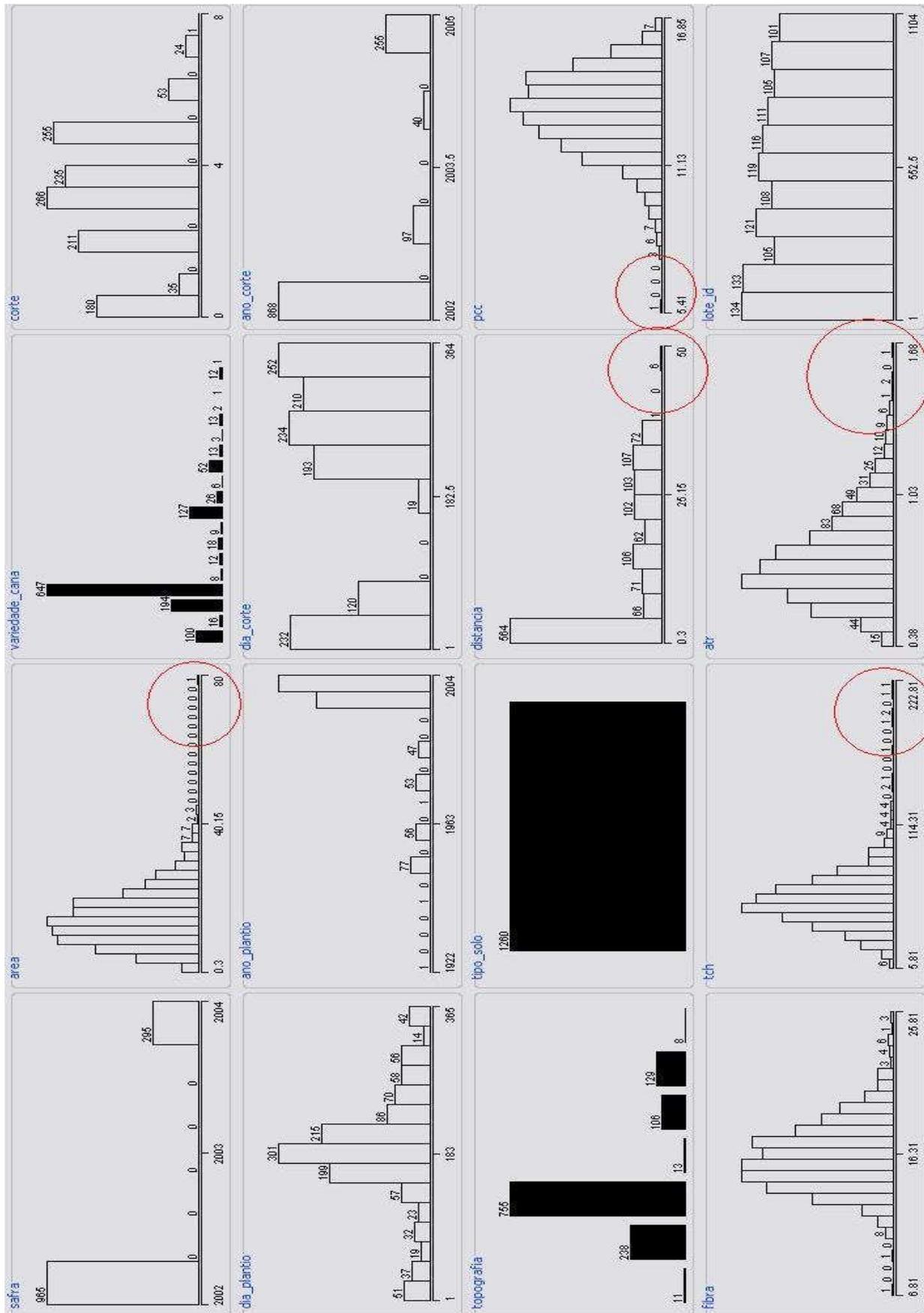


Figura 5.3. Base de dados CLIENTE_2 antes do pré-processamento: atributos e distribuição.

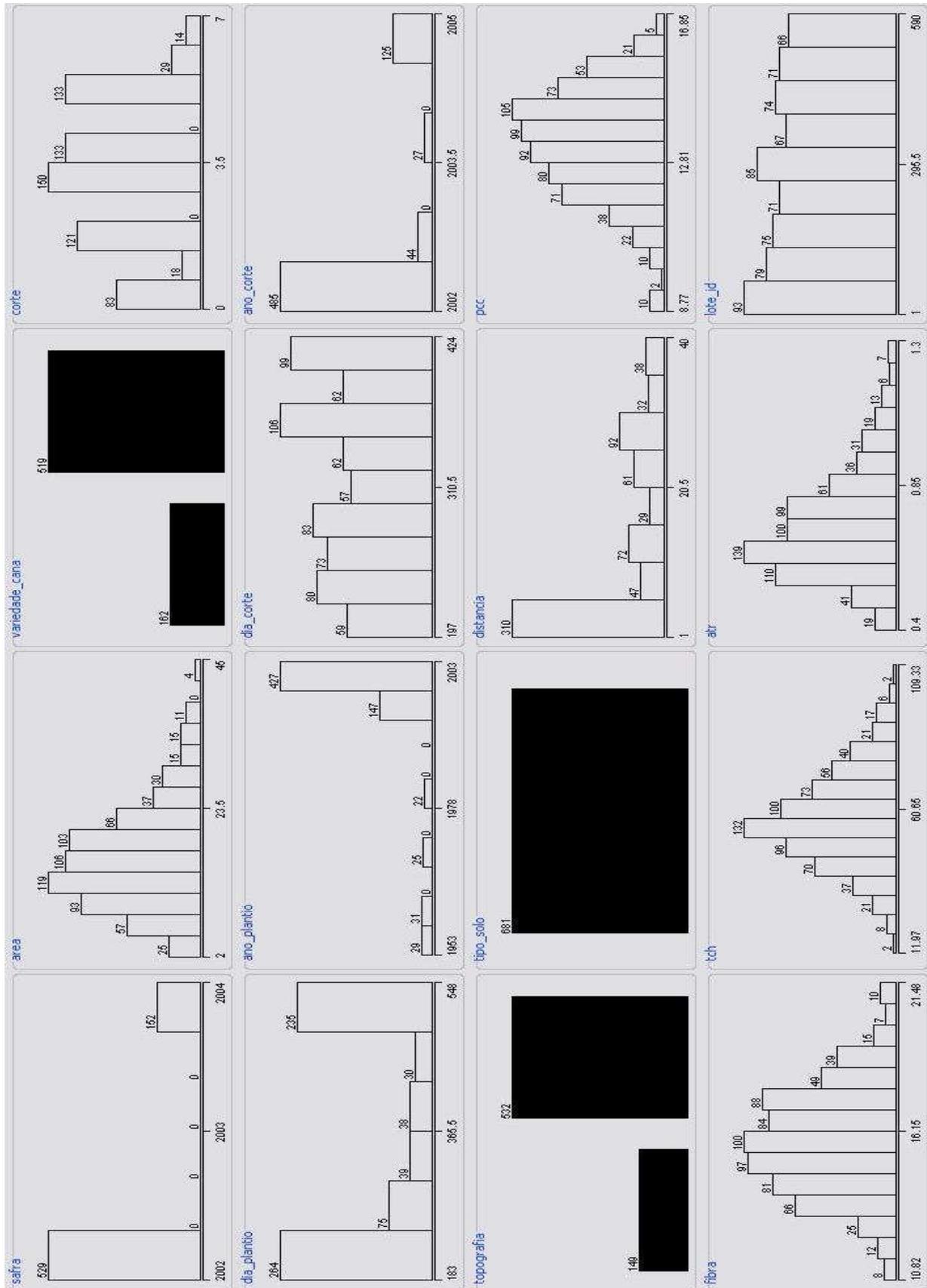


Figura 5.4. Base de dados CLIENTE_2 após o pré-processamento.

5.2 Modelagem Neural

Para cada uma das bases de dados utilizadas, foi necessário modelar uma RNA diferente. Todas as modelagens foram determinadas experimentalmente, através de várias simulações realizadas com o simulador SNNS (do inglês, *Stuttgart Neural Network Simulator*) [Stuttgart2008] e com a ferramenta Intelicolheita [Pacheco2006]. Observe-se que as bases de dados são utilizadas em dois momentos distintos da proposta: na predição (RNA) e na otimização (EMOO). Enquanto que para treinar, validar e testar a modelagem proposta para RNA são utilizados todos os padrões disponíveis, a parte EMOO da proposta emula o ambiente real de decisão de modo que trabalha com um conjunto completo de lotes estimados da usina em questão. Esta pequena diferença ficará mais clara nas seções seguintes.

5.2.1 Modelagem neural para CLIENTE_0

A rede utilizada na *Fase Um* e portanto com a base de dados CLIENTE_0 foi a mesma definida anteriormente em [Pacheco2005] e utilizada nos demais trabalhos [Pacheco2006], [Oliveira2006] e [Oliveira2007]. A Tabela 5.3 mostra a melhor topologia encontrada, a configuração paramétrica utilizada e as magnitudes médias de erro relativo obtidas (do inglês, *mean magnitude of relative error*, MMRE).

Tabela 5.3. Topologia, configuração e MMRE da RNA usada com a base CLIENTE_0.

Número de neurônios			Parâmetros			MMRE de predição (%)		
entrada	escondida	saída	η	ciclos	função de ativação	PCC	FIBRA	TCH
38	16	3	0.1	15000	Logística	4.38	23.94	7.48

É importante observar que a base CLIENTE_0 não possui atributos que identifiquem os lotes e desta forma, após o treinamento, apenas os 418 padrões do arquivo de testes puderam ser utilizados. Cria-se uma numeração nestes padrões de modo que todos os trabalhos baseados nesta base trabalharam com um modelo de usina com 418 lotes disponíveis para a colheita.

5.2.2 Modelagem neural para CLIENTE_1 e CLIENTE_2

Como estas bases foram disponibilizadas simultaneamente durante este trabalho, as pesquisas e simulações a cerca da melhor representação seguiram as mesmas motivações. Por isso nesta mesma seção estão descritos os objetivos a serem alcançados para CLIENTE_1 e CLIENTE_2.

Como visto anteriormente, uma rede neural artificial é um aproximador de funções de modo que diferentes escalas de valores podem contribuir de maneiras distintas (positivamente ou

negativamente). Outro fator relevante tratado pelo pré-processamento foi a representação de valores não-quantitativos ou qualitativos. Por exemplo, para representar as variedades de cana-de-açúcar não se deve usar números identificadores o que poderia sugerir um efeito indesejado de ordem, onde uma codificação unária separa melhor as classes.

Em linha com essa investigação inicial de modelos foram testados as seguintes hipóteses para pré-processamentos. De (a) até (i) são informações usadas nas entradas das redes neurais e a partir de (j), usadas na saída das redes neurais.

- a) Safra: normalização entre zero e um a partir da ordem cronológica;
- b) Variedade da cana-de-açúcar: codificação unária. Quatro variedades em CLIENTE_1 e duas variedades em CLIENTE_2 (ver Figura 5.2 e Figura 5.4);
- c) Estágio da maturação (*i.e.* número do corte): normalização entre zero e um;
- d) Dia do plantio e da colheita: três abordagens foram testadas:
 - i. Codificação binária da quinzena – apenas um bit, 0 ou 1 representando a primeira ou segunda quinzena, respectivamente;
 - ii. Codificação unária – dois bits, ativados isoladamente para cada quinzena;
 - iii. Normalização considerando o dia no ano. Essa abordagem intrinsecamente embute a representação do mês. Observe que esta é a forma representada na Figura 5.1, Figura 5.2, Figura 5.3 e Figura 5.4.
- e) Mês do plantio e da colheita: duas abordagens testadas.
 - i. Diretamente com codificação unária – doze bits;
 - ii. Indiretamente representado através do dia no ano (item anterior).
- f) Ano do plantio e da colheita: normalização entre zero e um;
- g) Topografia: codificação unária – dois tipos em ambas as bases (ver Figura 5.2 e Figura 5.4);
- h) Tipo do solo: em ambas as bases, após a análise inicial, restou apenas um tipo de solo, de modo que não foi necessária a sua representação – foi considerado apenas o tipo de solo mais frequente;
- i) Área, distância e *lote_id*: como são irrelevantes no modelo de estimativa de maturação da cana-de-açúcar proposto, não foram considerados na modelagem neural. Entretanto foram utilizados pelo módulo otimizador que não precisa de pré-processamento.
- j) PCC (médio): normalização entre zero e um;
- k) TCH (médio): normalização entre zero e um;

l) FIBRA (média): normalização entre zero e um;

m) ATR (médio): normalização entre zero e um.

Os pré-processamentos resultaram em três diferentes abordagens para cada uma das bases de dados. As diferenças e o número de bits utilizados em cada uma delas estão apresentados na Tabela 5.4.

Tabela 5.4 Diferentes configurações de pré-processamento confrontadas nas bases de dados CLIENTE_1 e CLIENTE_2 e as respectivas quantidades de bits necessárias.

Base	Safra	varie- dade	Num. corte	Plantio			Corte			top.	PCC	TCH	FIBRA	ATR	Total bits
				dia	mês	ano	dia	mês	ano						
CLIENTE_1	1	4	1	1	12	1	1	12	1	2	1	1	1	1	40
	1	4	1	2	12	1	2	12	1	2	1	1	1	1	42
	1	4	1	1	0	1	1	0	1	2	1	1	1	1	16
CLIENTE_2	1	2	1	1	12	1	1	12	1	2	1	1	1	1	38
	1	2	1	2	12	1	2	12	1	2	1	1	1	1	40
	1	2	1	1	0	1	1	0	1	2	1	1	1	1	14

Diversas simulações foram realizadas na tentativa de encontrar uma topologia que melhor estimasse os indicadores agrônômicos PCC, TCH, FIBRA e ATR. Através de um planejamento fatorial se variou o número de neurônios na camada escondida, a taxa de aprendizado (η) e o número de ciclos. Testou-se também como a RNA se comporta quando realiza a predição dos indicadores isoladamente ou com diferentes combinações. Assim, foram testadas (i) redes que realizavam a predição dos quatro indicadores simultaneamente, (ii) redes que estimavam cada um dos quatro indicadores isoladamente e (iii) redes que combinavam os indicadores a serem preditos.

Os melhores resultados para cada pré-processamento estão exibidos em negrito na Tabela 5.5. Os melhores resultados para cada uma das saídas foram alcançados por configurações diferentes. As abordagens que utilizam apenas 1 bit para representar o dia no ano, descartando 12 bits por data referentes ao mês, reduzem o total de bits em mais de 50%, o que reduz a complexidade e o espaço para armazenamento da rede. Ainda assim, esta abordagem conseguiu resultados superiores ou bastante similares aos obtidos pelas melhores representações. Isto sugere que a abordagem utilizada anteriormente não estava adequada para o problema. Desta forma, optamos por este pré-processamento nas *Fases Dois e Três*.

Ao contrário da base CLIENTE_0, as bases CLIENTE_1 e CLIENTE_2 contêm identificadores para os lotes, de modo que as estimativas podem ser realizadas sobre safras completas. Assim o módulo otimizador recebe:

- Da base CLIENTE_1 a predição de 937 lotes referentes à safra de 2006 (os dados disponíveis eram de 2001 – 2005);
- Da base CLIENTE_2 a predição de 590 lotes referentes à safra de 2005 (os dados disponíveis eram de 2002 e 2004).

Finalmente, após as predições, um pós-processamento é realizado para que os atributos utilizados pela RNA retornem aos seus formatos originais (*i.e.* suas escalas). Assim, é possível a interação entre o decisor e os outros módulos seguintes do sistema.

Tabela 5.5. Melhores resultados de predição e parametrização para diferentes abordagens aplicadas às bases CLIENTE_1 e CLIENTE_2.

Diferentes redes	CLIENTE_1				CLIENTE_2			
	1	2	3	4	1	2	3	4
Atributos da predição	PCC, FIBRA e ATR.	TCH	PCC, FIBRA, ATR e TCH.	PCC, FIBRA, ATR e TCH.	PCC, FIBRA e ATR.	TCH	PCC, FIBRA, ATR e TCH.	PCC, FIBRA, ATR e TCH.
nº entradas	36	36	38	12	34	34	36	10
nº escondidos	6	3	4	6	6	3	4	10
nº saídas	3	1	4	4	3	1	4	4
nº neurônios	45	40	46	22	43	38	44	24
η	0.1	0.1	0.05	0.05	0.1	0.1	0.05	0.05
nº ciclos	11k	10.3k	60k	40k	15k	17.4k	10k	20k
Tempo (s)	123	61	610	300	55	36	37	67
MMRE PCC (%)	7.57	-	7.86	7.89	6.94	-	8.9	6.95
MMRE FIBRA (%)	7.7	-	8.25	7.6	6.59	-	11.6	7.1
MMRE ATR (%)	6.48	-	6.77	6.8	16.21	-	18.59	15.52
MMRE TCH (%)	-	32.3	31.84	27.25	-	18.02	22.52	21.14

5.3 Resultados da Evolução da Proposta

Com a disponibilização das bases de dados e com os pré-processamentos realizados, as contribuições a cerca das decisões de colheita puderam ser executadas. Detalharemos a seguir os experimentos realizados e os resultados obtidos, agrupados em fases, do mesmo modo em que a contribuição foi apresentada no capítulo anterior.

5.3.1 Resultados da Fase Um

Na seção 4.3.1, descreveu-se como a formulação multi-objetiva foi pioneiramente introduzida no problema da colheita de cana-de-açúcar. Os três macros objetivos estabelecidos foram: (i) definir uma formulação MO para o problema, (ii) confrontar algoritmos MO clássicos e evolucionários e (iii) realizar uma avaliação ao nível de safra e não apenas em dias isolados de colheita.

Duas abordagens foram definidas de acordo como a restrição de tonelagem colhida era tratada: (a) a primeira que utilizava uma função de penalidade e (b) a segunda que transformava a restrição em mais um objetivo. Todos os experimentos foram realizados com a base CLIENTE_0, logo com informações referentes a 418 lotes. Para tanto, assumiu-se o seguinte cenário hipotético:

- Todos os lotes devem ser colhidos. Como mencionado, os trabalhos anteriores avaliavam um único dia de colheita, mas agora todo um ano produtivo será considerado;
- A tonelagem mínima desejada foi fixada para 4000 toneladas por dia de colheita (por arquivo de predição). Isto significa que foi assumido que a usina precisaria desta tonelagem, diariamente, para continuar trabalhando continuamente e que ela não conseguiria moer muito mais além disto. A heurística adotada no módulo decisor foi de uma tolerância de -0.5% a +5% na tonelagem desejada. Portanto, os limites aceitos foram de 3980 até 4200 toneladas;
- Devido às pequenas variabilidades nos indicadores agronômicos entre dias consecutivos (mesma abordagem utilizada no pré-processamento), apenas duas predições são realizadas por mês, ou seja, uma a cada quinzena;
- A colheita deve ser finalizada (todos os lotes colhidos) em no máximo 12 iterações. Isto corresponde ao período médio de 6 meses de colheita num ano;
- Se na décima segunda iteração o limite de 4200 toneladas for atingido e ainda existirem lotes para serem colhidos, todos eles deverão ser selecionados para colheita.

A. Representação dos Indivíduos

Os indivíduos usados no NSGA-II representavam os lotes disponíveis nos campos (*i.e.* lotes ainda não colhidos). Utilizou-se uma representação binária (*bit string*), onde *0* indica “não selecionado para colheita” e *1* representa o contrário. A Figura 5.5 ilustra um indivíduo quando existem 40 lotes disponíveis e o sistema sugere 15 para serem colhidos.

A cada iteração (*i.e.* dia de colheita) um novo mapeamento é implementado entre os índices dos indivíduos da população e os identificadores dos lotes disponíveis. Isto permite que todos os lotes sejam colhidos e selecionados apenas uma única vez durante toda a safra. É importante observar que enquanto a colheita de toda safra não é finalizada, novos dias de colheita são realizados, de modo que os lotes selecionados serão colhidos. Conseqüentemente, o número de lotes disponíveis diminui assim como o tamanho do indivíduo (do AG).

Genes de um indivíduo (*i.e.* lotes disponíveis) →
1001001000 1110000010 1000110001 0010101001

Figura 5.5. Indivíduo que representa a sugestão de colheita de 15 lotes entre os 40 disponíveis.

B. Configuração dos Parâmetros

Os parâmetros utilizados no NSGA-II foram encontrados experimentalmente. Em particular, um novo critério de parada foi usado paralelamente ao número de iterações. Ele representa quantas vezes a população é totalmente formada por soluções não-dominadas (primeiro nível de Pareto *front*). Quando este atributo assumiu valores maiores que um, os experimentos sempre finalizavam com o número máximo de gerações, mas na medida em que este valor decresceu, as simulações foram mais curtas e produziram melhores resultados.

A Tabela 5.6 ilustra o melhor conjunto paramétrico encontrado de acordo com os experimentos. Verificou-se experimentalmente qual operador de cruzamento estaria melhor adaptado ao problema e após testes com cruzamentos de um-ponto, de dois-pontos e uniforme, este último mostrou alcançar melhores resultados.

A população inicial foi formada por indivíduos gerados aleatoriamente de acordo com a disponibilidade dos lotes no campo que era inicialmente 418 lotes. Cada bit nos genótipos dos indivíduos tinha 50% de chance de ser ativado.

Tabela 5.6. Melhor configuração paramétrica encontrada para o problema abordado na *Fase Um*.

Parâmetro	Valor
Taxa de cruzamento	90%
Taxa de mutação	5%
Tamanho da população	100
Número de gerações	90
Comprimento inicial dos indivíduos	418
Toda população no 1º Pareto frente	1

C. Resultados dos Experimentos

O algoritmo NSGA-II foi sucessivamente aplicado nas duas abordagens testadas e em ambas as situações a população conseguiu alcançar o Pareto *front*. Entretanto a abordagem *b* obteve uma melhor diversidade que a abordagem *a*, e ainda alcançou os melhores valores de produção.

Por questões de implementação e utilizando o conceito de dualidade, os objetivos que naturalmente são maximizações foram convertidos em minimizações (*i.e.* produções de PCC e FIBRA) de modo que nas duas abordagens aplicadas todos os objetivos foram minimizações. Note que o terceiro objetivo criado para a abordagem *b*, por natureza, já é uma minimização e assim nenhuma alteração foi necessária. Logo, os valores negativos apresentados na Figura 5.6 e

na Figura 5.7 são apenas convenções. Obviamente, não é possível conseguir produções de PCC ou FIBRA negativas.

Na Figura 5.6 pode se observar toda a convergência realizada pelo MOEA tanto na abordagem *a* (dois objetivos) quanto com a abordagem *b* (três objetivos). As estrelas pretas representam as populações iniciais geradas aleatoriamente enquanto que os círculos brancos representam o Pareto *front* formado após a otimização. Note que a diferença entre a população inicial e final é imensa em ambas as situações. Isto pode indicar também que o espaço de decisão é muito vasto, ou seja, o número de combinações possíveis é extremamente elevado o que revela a complexidade do problema.

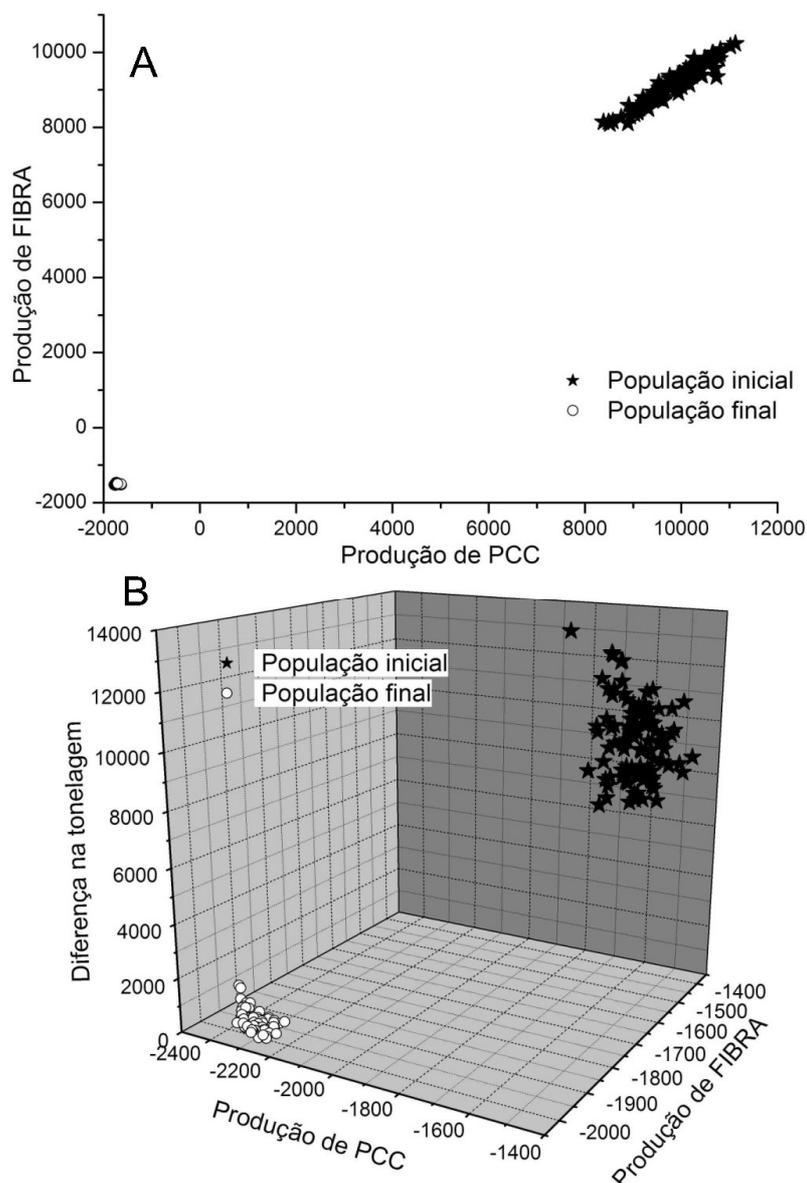


Figura 5.6. Convergência das populações após otimização multi-objetivo nas abordagens: *a* (A) e com a abordagem *b* (B).

A Figura 5.7 ilustra a superfície criada pela última geração, ou seja o Pareto *front* formado para as duas abordagens. Apesar dos bons resultados, a diversidade de soluções encontradas ainda foi baixa. Na abordagem *b*, que apresentou melhor diversidade, observa-se que a transformação da restrição do problema em um terceiro objetivo melhora os resultados finais; entretanto, e conseqüentemente, apresenta soluções com uma dispersão maior entre a tonelagem desejada e a tonelagem fornecida. Por outro lado, as poucas soluções encontradas com a abordagem *a* garantem limites aceitáveis na tonelagem desejada.

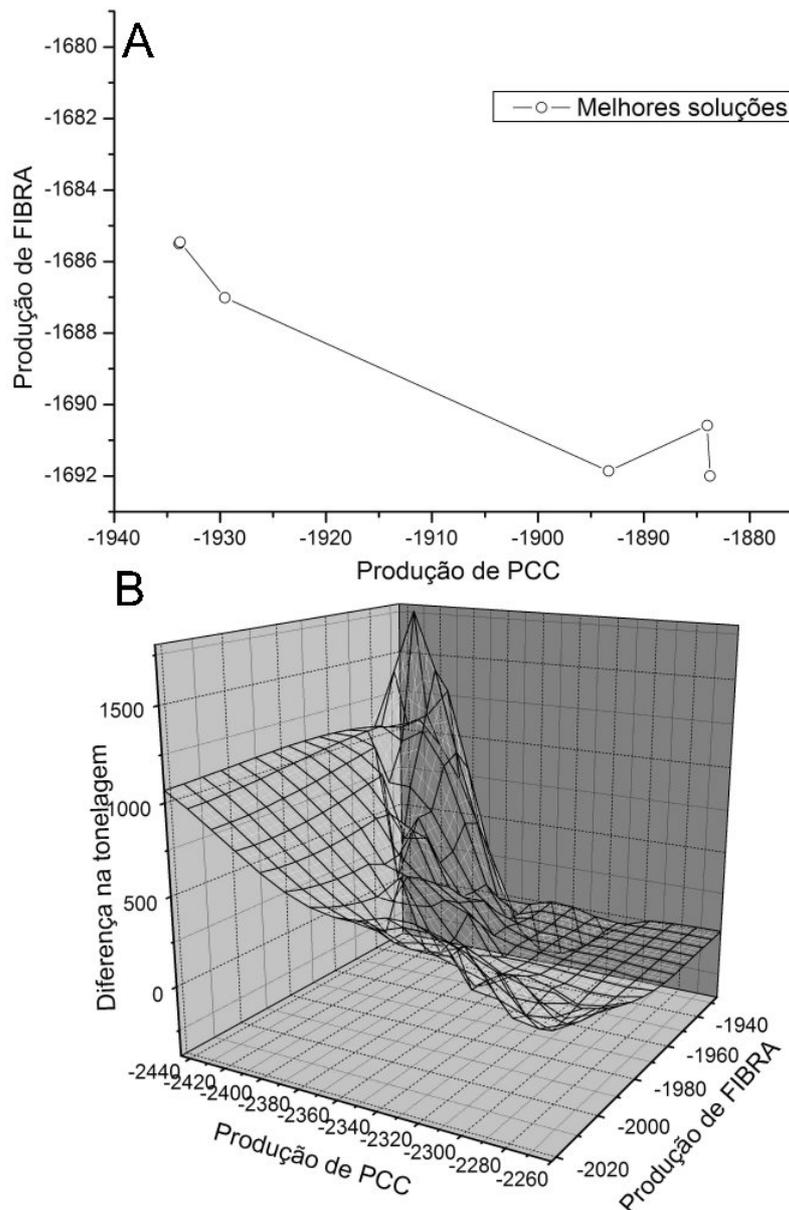


Figura 5.7. Superfícies do Pareto frente obtidos com a abordagem *a* (A) e com a abordagem *b* (B).

Observe que apesar de estarem próximas no Pareto formado, duas soluções vizinhas podem ser bastante diferentes. Isto ocorre porque o número de combinações possíveis entre os lotes a serem colhidos é enorme. A Figura 5.8 ilustra este fato. Nela, duas soluções encontradas (vizinhas no Pareto *front*) foram decodificadas e foram representadas visualmente a partir da coloração dos lotes selecionados para a colheita. Um terceiro bloco sobrepõe estas soluções colorindo de verde os lotes selecionados por ambas as soluções. Quanto mais pontos verdes são vistos neste bloco, mais similares são as soluções em questão. No entanto, apesar de semelhantes no Pareto, as duas soluções são, de fato, duas colheitas bastante distintas.

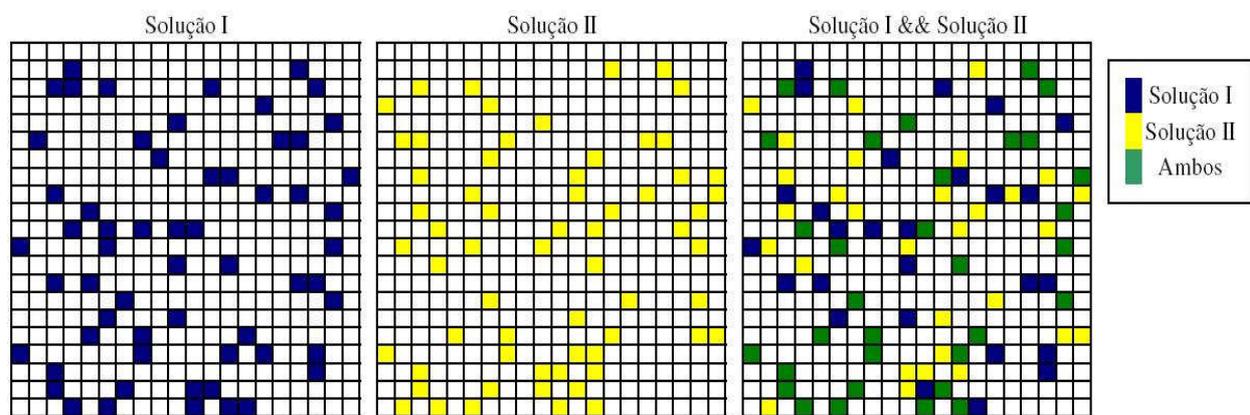


Figura 5.8. Decodificação de duas soluções encontradas pela abordagem *b* (blocos da esquerda e central). Cada quadrado colorido indica que o lote foi escolhido para a colheita: na cor azul pela solução I e na cor amarela pela solução II. O bloco mais a direita sobrepõe as duas soluções e os lotes selecionados por ambas aparecem de verde.

D. Comparação com Outras Técnicas

Com a finalidade de assegurar a eficiência do método proposto e da formulação de aplicação, compararam-se os resultados obtidos com o algoritmo NSGA-II com outros métodos clássicos utilizando a mesma base de dados e o mesmo cenário hipotético.

A Tabela 5.7 apresenta os valores obtidos nas produções finais de cana-de-açúcar, de PCC e de FIBRA para os melhores métodos aplicados. Estes resultados foram obtidos a partir da média de 10 execuções, todos com desvio padrão menor que 1% para cada uma das três saídas.

Como esperado, as abordagens evolucionárias *a* e *b* (*i.e.* com dois e três objetivos, respectivamente) foram mais eficazes que as tentativas clássicas. Usando o método Lexicográfico, por exemplo, a produção de cana-de-açúcar foi, aproximadamente, de 46 mil

toneladas enquanto que a abordagem *b* (NSGA-II) conseguiu 52 mil toneladas. Em termos práticos, isto poderia significar uma diferença de mais de Int\$⁹124 mil.

Tabela 5.7. Valores das produções de cana-de-açúcar, PCC e FIBRA alcançadas pelos melhores métodos clássicos e evolucionários aplicados no problema.

Técnica	Produção de		
	Cana	PCC	FIBRA
NSGA-II (3 objs)	52644.65	778419.15	740252.40
NSGA-II (2 objs)	51337.62	756915.75	727215.22
AHP	47520.80	701325.18	701280.50
MAUT	47395.93	700453.01	700079.16
Lexicográfico	46642.91	692220.13	690656.33

A Figura 5.9 ilustra a superioridade do melhor algoritmo aplicado – NSGA-II configurado com 3 objetivos – comparado com os demais. A figura exibe o melhor método em 1.00 representando 100% da máxima produção de cana-de-açúcar obtida, enquanto os demais métodos assumem seus respectivos valores proporcionais. Assim o melhor representante clássico, o método AHP, ainda foi 10% inferior ao NSGA-II.

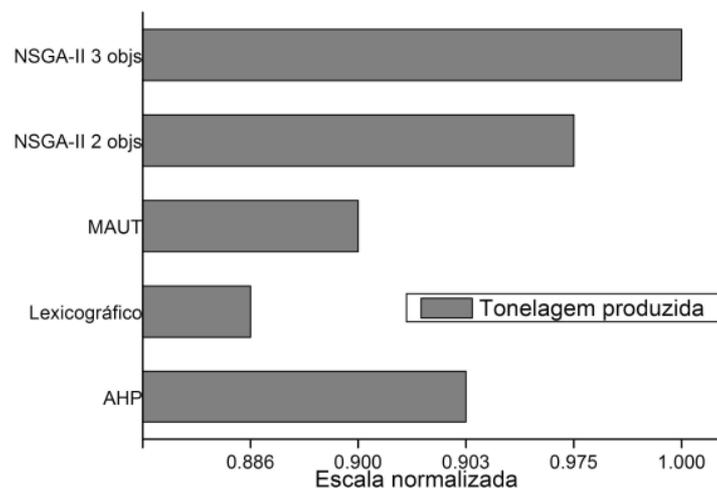


Figura 5.9. Uma comparação relativa entre a tonagem de cana-de-açúcar produzida pelas diferentes técnicas de otimização multi-objetivo aplicadas na *Fase Um* desta pesquisa.

5.3.2 Resultados da Fase Dois

Uma vez que o problema da colheita de cana-de-açúcar foi reformulado, e com a comprovação de uma aplicabilidade eficiente e até superior dos algoritmos evolucionários, em particular o NSGA-

⁹ Int\$ - Preço internacional. De acordo com a FAO a cana-de-açúcar custa, aproximadamente, Int\$20.77 por tonelada. Estes valores foram calculados baseados nos preços internacionais entre 1999 e 2001 [FAO2007].

II sobre os algoritmos clássicos, tem-se o início da segunda fase. Os objetivos desta fase foram: (i) combinar formas de representação para o problema, operadores genéticos e algoritmos evolucionários para descobrir como o problema é mais bem solucionado; e (ii) incorporar dados logísticos à otimização. Mais detalhes na seção 4.3.2.

Nesta seção são apresentados os resultados experimentais da contribuição proposta sobre duas bases de dados testadas. Em adição, comenta-se sobre as métricas utilizadas nas comparações e os parâmetros utilizados nas simulações.

A. Métricas de Desempenho

Como os objetivos desta fase são diferentes dos estabelecidos na fase anterior, conseqüentemente, as métricas utilizadas nesta avaliação devem ser diferentes também. Anteriormente, desejava-se aumentar a produção final de toda a safra, mas conseqüentemente, os resultados obtidos apresentavam pouca diversidade de soluções em termos dos múltiplos objetivos. Agora, retorna-se à tentativa de melhoria nos resultados multi-objetivos de um único dia, o que implicaria, portanto, numa melhoria ao final de todo o processo. As abordagens utilizadas aqui são aplicadas apenas no primeiro dia de colheita – o mais complexo devido ao maior número de combinações disponíveis de lotes para colheita.

Existem diversas métricas de desempenho que vêm sendo utilizadas em comparações EMOO mas não há consenso sobre qual delas seria a melhor [Deb2001]. Fonseca *et al.* sugere algumas métricas unárias e outras binárias para serem aplicadas [Fonseca2005], mas aqui aplicamos a combinação das três métricas utilizadas nos principais trabalhos inspiradores desta fase, [Mumford2003] e [Colombo2005]. Estas métricas não avaliam soluções isoladas, mas sim os conjuntos de aproximação obtidos com as otimizações. As métricas escolhidas foram:

- Hipervolume (a métrica S) – que a partir do cálculo de hipercubos, mede quanto do espaço de busca foi explorado. Em outros termos, ela mede a proximidade com o Pareto *front*. Quando a população se aproxima do Pareto ideal, mais próximo de 1 será o valor do hipervolume [Zitzler1998];
- *Generalized Spread* (espalhamento) – mede quão bem estão distribuídas as soluções encontradas sobre o Pareto. Ou seja, calcula como o conjunto de aproximação contempla todos os objetivos considerados na otimização, como as soluções conseguem ser representativas para todos eles. Quando o conjunto de aproximação apresenta um espalhamento perfeito, *generalized spread* medirá 0 [Zhou2006];

- Tempo de execução – mede o desempenho das diferentes abordagens utilizadas quando realizam exatamente o mesmo número de chamadas às funções de cálculo de objetivo implementados sob o mesmo framework. O tempo de execução foi medido em um PC Intel Core 2 Duo CPU 6400 @ 2.13GHz com 2GB de RAM, utilizando a versão 3.3.0 da plataforma Eclipse apoiada pela máquina virtual Java de versão 1.6.0_03.

B. Parâmetros das Simulações

As simulações foram realizadas utilizando as bases de dados reais de duas usinas, CLIENTE_1 e CLIENTE_2. Para que houvesse justiça na comparação, aplicou-se a mesma configuração paramétrica para todos os algoritmos em cada um dos problemas definidos para as duas bases de dados. Isto inclui: tamanho da população, taxa de cruzamento, taxa de mutação, número máximo de chamadas às funções objetivo, entre outros.

Os valores utilizados para cada atributo foram encontrados experimentalmente. O tamanho da população foi proporcional ao tamanho da usina (número de lotes), bem como o número máximo de avaliações. As diferentes tonelagens desejadas representam a condição restritiva das mochilas (capacidade) de, aproximadamente, 10% e 30% para as bases CLIENTE_1 e CLIENTE_2 respectivamente. A Tabela 5.8 apresenta toda a configuração paramétrica para os dois problemas projetados.

Tabela 5.8. Parâmetros utilizados em ambas as bases de dados nos experimentos realizados na *Fase Dois*.

Parâmetros	CLIENTE_1	CLIENTE_2
Número de lotes (n)	937	590
Tamanho da população	400	200
Tonelagem desejada	50000	150000
Número de avaliações	80000	25000
Taxa de cruzamento		0.9
Taxa de mutação		$1 / n$
Tamanho do arquivo ¹⁰	400	200
Bi-sections ¹¹		5
Operador de seleção	Torneio Binário	

C. Resultados

Os resultados das 12 abordagens utilizadas são apresentados e organizados por métrica de desempenho, base de dados e algoritmo aplicado. Todas as abordagens (*i.e.* combinações de

¹⁰ Não é aplicado ao NSGA-II.

¹¹ O número máximo de bi-divisões para o gride adaptativo (PESA-II).

algoritmos e representações do problema) são ilustradas com gráficos *box plot* com 30 simulações. As 12 abordagens testadas foram:

- NSGA-II: (i) representação binária – *N.bit*; (ii) binária com a heurística *best fit* – *N.bitD*; (iii) representação simbólica com cruzamento em ciclo – *N.cx*; e, (iv) representação simbólica com o cruzamento *partially matched* – *N.pmx*;
- SPEA2: (i) *S.bit*; (ii) *S.bitD*; (iii) *S.cx*; e, (iv) *S.pmx*;
- PESA-II: (i) *P.bit*; (ii) *P.bitD*; (iii) *P.cx*; e, (iv) *P.pmx*.

I) Resultados para Hipervolume (HV)

Os resultados obtidos para a métrica HV são exibidos na Figura 5.10 e na Figura 5.11, para as bases CLIENTE_1 e CLIENTE_2, respectivamente. Apesar dos gráficos de HV estarem com diferentes escalas em ambas as bases (as melhores aproximações em CLIENTE_1 estão entre 0.91 e 1; e em CLIENTE_2 entre 0.97 e 1), um padrão comum por algoritmos entre as bases de dados é revelado.

Todos os HV das representações aplicadas sob o algoritmo NSGA-II, na média, foram levemente superiores aos obtidos pelas respectivas representações utilizando o algoritmo SPEA-II. Entretanto, ao considerarmos na análise os desvios destas distribuições, percebe-se que estes dois algoritmos obtiveram desempenhos equivalentes em todas estas representações. Quando a comparação é realizada no nível de representações, nota-se que as abordagens *cx* e *bit* estiveram mais próximas do pareto ideal, seguidas pelas abordagens *pmx* e *bitD*. Com a base CLIENTE_1, onde o problema foi definido de modo mais complexo, as representações *cx* conseguem HV ligeiramente melhores (na média) que as representações *bit*, mas não o suficiente para ser declarada como “a melhor” representação.

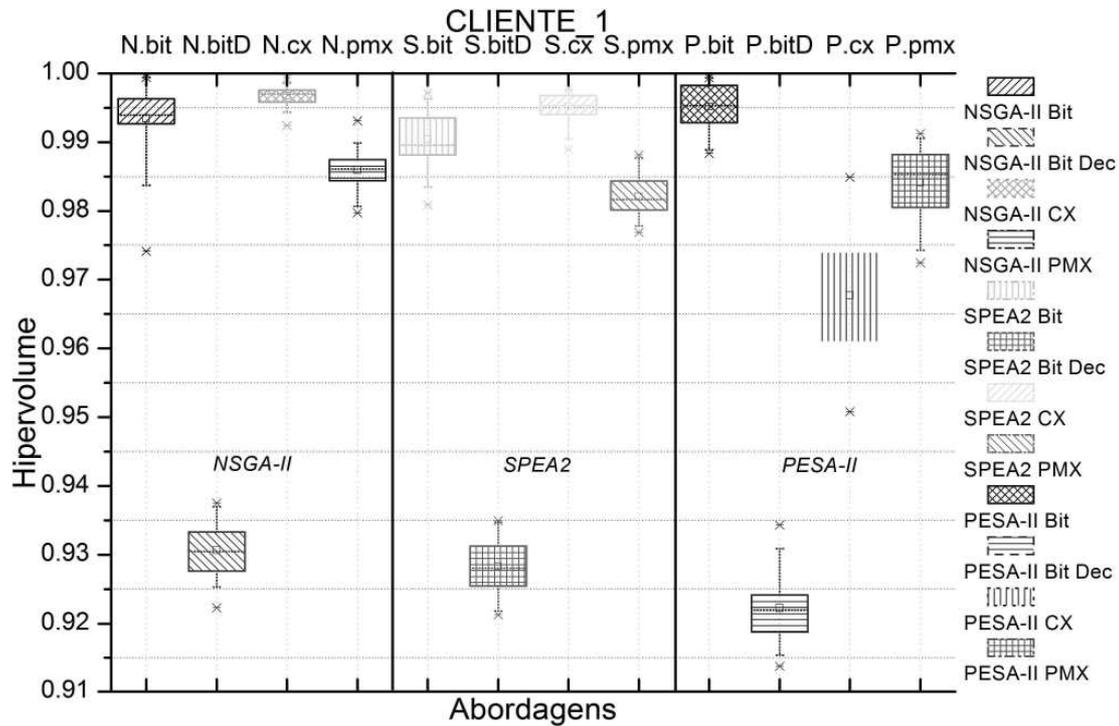


Figura 5.10. Hipervolume para as diferentes abordagens com a base CLIENTE_1.

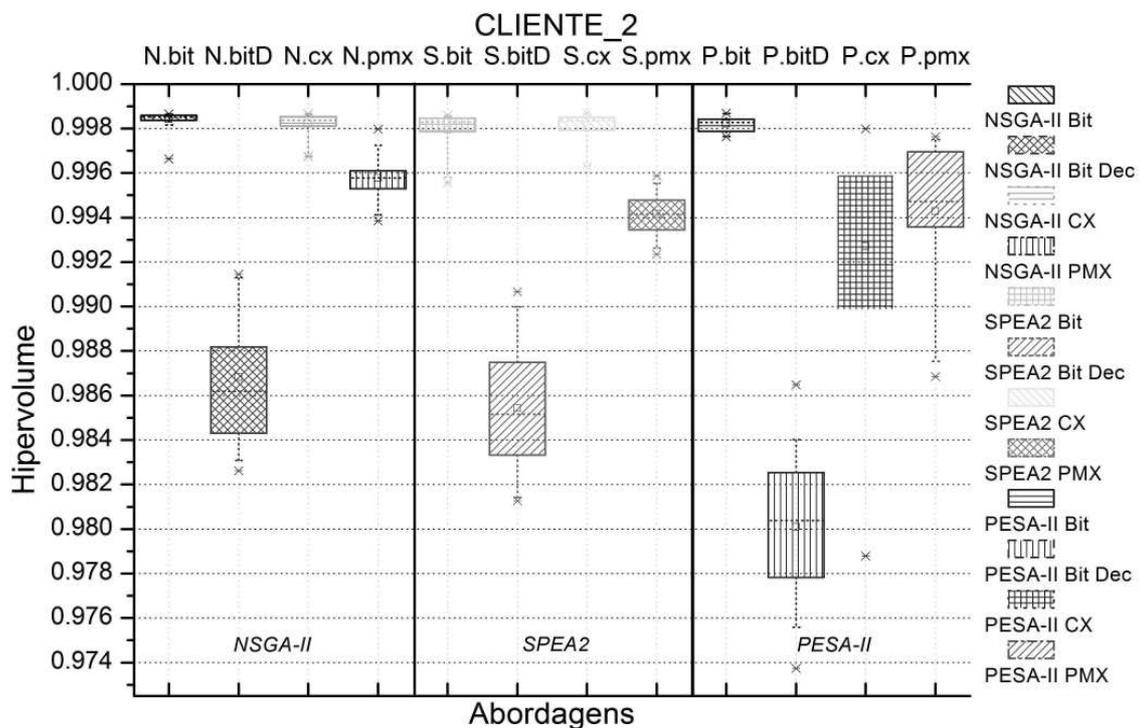


Figura 5.11. Hipervolume para as diferentes abordagens com a base CLIENTE_2.

Já o algoritmo PESA-II apresentou resultados um pouco diferentes dos demais algoritmos quando se compara a classificação das representações intra-algoritmos. As representações *bit*, *pmx*, *cx* e *bitD* foram sistematicamente, nesta seqüência, as melhores representações para este algoritmo; com exceção das abordagens *pmx* e *cx* na base CLIENTE_2 onde *cx* apresenta uma

grande dispersão e, portanto, pode ser considerada equivalente a *pmx*. Quando a comparação é inter-algoritmos, percebe-se equivalência nas representações *bit* e *pmx*; *P.bitD* é um pouco inferior as demais, mas é a abordagem *cx* quem mais perde desempenho; ela decai do 1º lugar em NSGA-II e SPEA2 para o 3º em PESA-II.

II) Resultados para Generalized Spread (GS)

Os resultados sobre a métrica que mede a diversidade entre as soluções (espalhamento) estão demonstrados na Figura 5.12 e na Figura 5.13 para as bases de dados CLIENTE_1 e CLIENTE_2, respectivamente. Novamente, assim como nos gráficos a cerca do hipervolume, percebe-se claramente um padrão comum entre as representações inter-algoritmos e intra-algoritmos.

O desempenho das representações intra-algoritmos é, do melhor para o pior, *pmx*, *cx*, *bitD* e *bit*. Mas devido à variância na amostra das simulações, por vezes, as representações *pmx* e *cx* são consideradas equivalentes, assim como as representações *bitD* e *bit*; a exceção é no algoritmo SPEA2 onde *S.pmx* é sempre superior a *S.cx*. Adicionalmente, ainda quando estes pares são considerados equivalentes (internamente), o primeiro (*pmx* e *cx*) é sempre superior ao segundo.

Quando se compara algoritmos, percebe-se que para ambas as bases de dados os algoritmos NSGA-II e PESA-II são equivalentes em todas as representações. Para o SPEA2, *bit* e *bitD* permanecem equivalentes aos demais algoritmos, no entanto, sua segunda melhor abordagem (*cx*) é inferior à segunda colocada nos demais. Assim, *S.pmx* (a melhor representação para SPEA2) é equivalente às segundas abordagens dos outros algoritmos (*N.cx* e *P.cx*).

III) Resultados para o Tempo de Execução (TE)

Os resultados da terceira métrica utilizada são exibidos na Figura 5.14 e na Figura 5.15 para as diferentes bases de dados.

Em CLIENTE_1 a melhor abordagem é *N.pmx* (cerca de 6 minutos), seguida bem próxima pelas soluções de *N.cx*, *S.pmx* e *P.pmx* (em torno de 7,5 minutos). Já as representações binárias (*bit* e *bitD*) demandaram muito mais tempo que as representações simbólicas, e consumiram aproximadamente 100 minutos em cada simulação.

Para a base CLIENTE_2 a maioria das abordagens executou em um razoável curto espaço de tempo, aproximadamente 4,5 minutos. Desta vez a exceção foi apenas para a representação binária *bit* que utilizou cerca de 12 minutos por simulação.

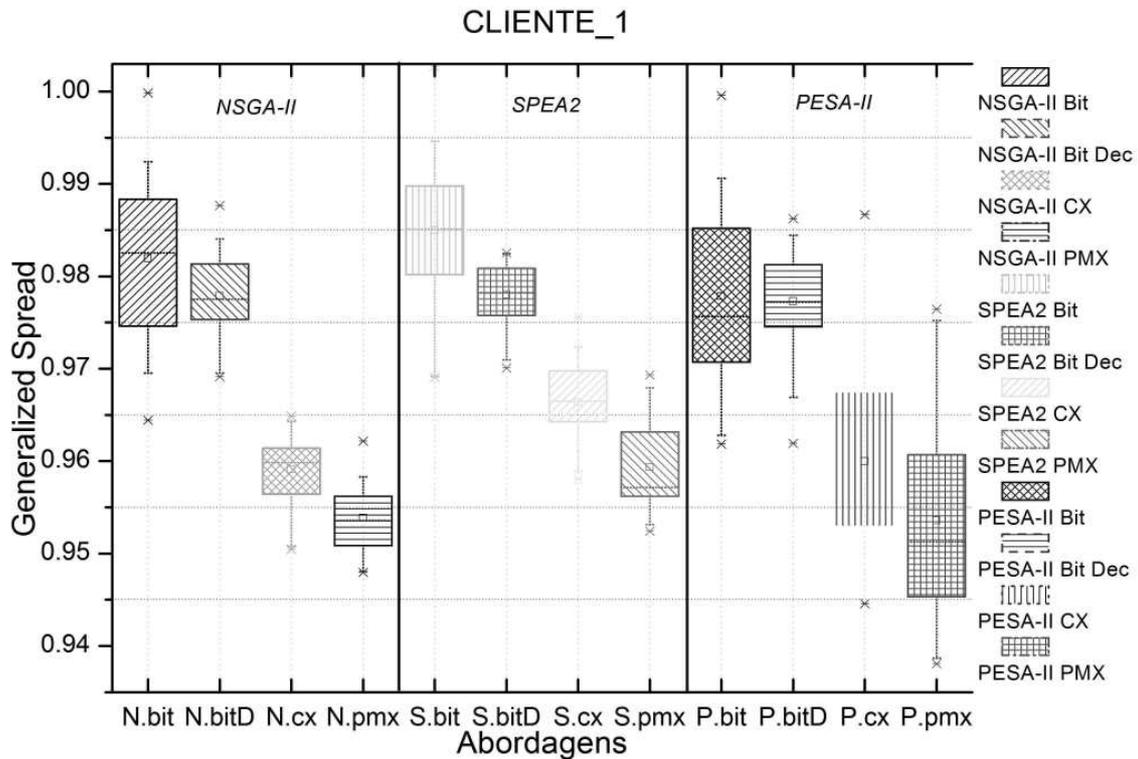


Figura 5.12. Espalhamento (GS) para as diferentes abordagens com a base CLIENTE_1.

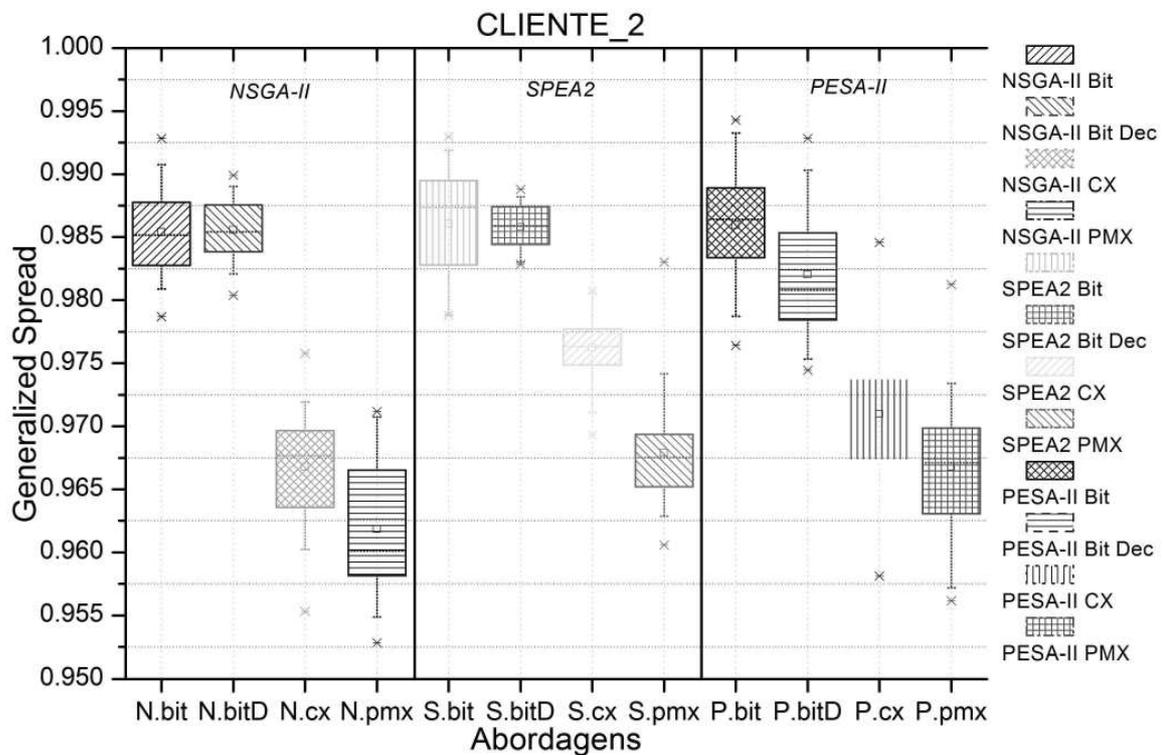


Figura 5.13. Espalhamento (GS) para as diferentes abordagens com a base CLIENTE_2.

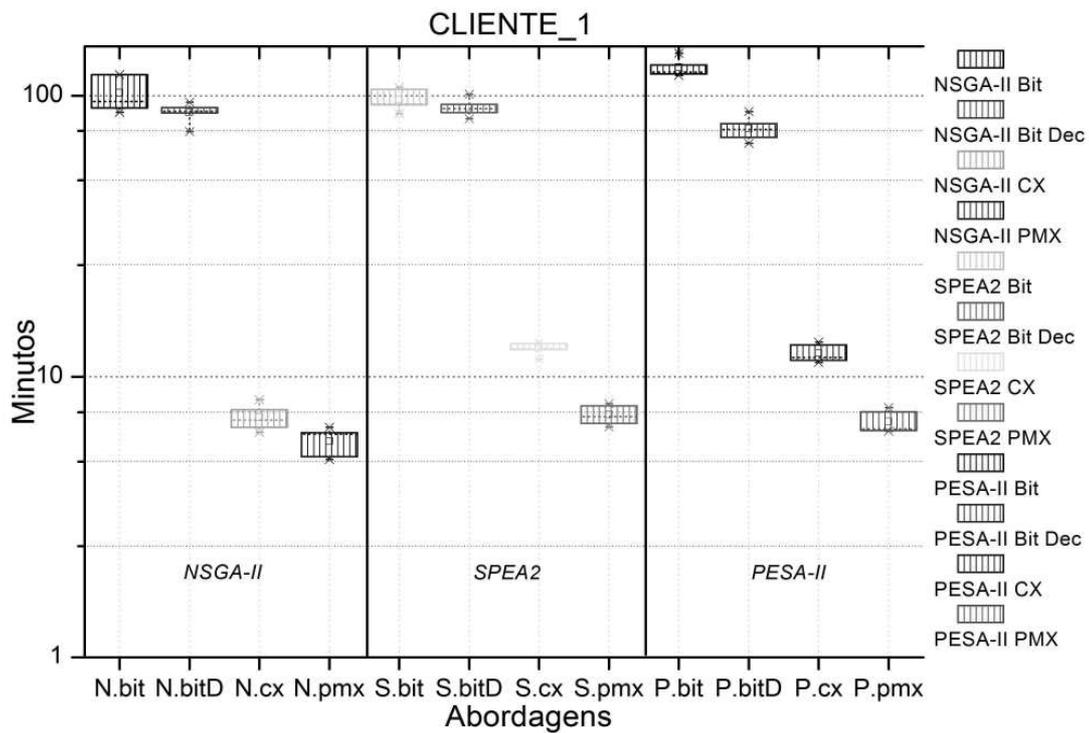


Figura 5.14. Tempos de execução para as diferentes abordagens utilizando a base CLIENTE_1.

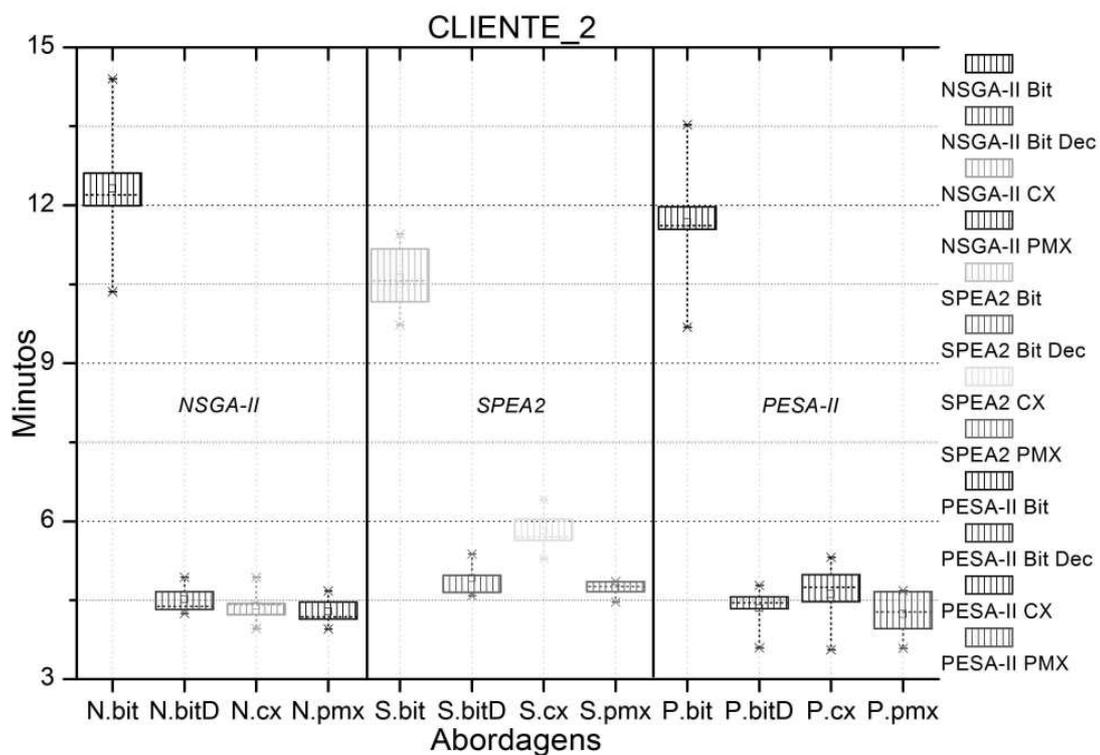


Figura 5.15. Tempos de execução para as diferentes abordagens utilizando a base CLIENTE_2.

É interessante notar como os tempos consumidos pelas duas bases de dados são de magnitudes tão diferentes. A base CLIENTE_1 apresenta quase o dobro de lotes da CLIENTE_2;

o problema para a primeira base foi projetado com o dobro da população e o triplo de restrições e de avaliações – tudo isso reflete diretamente nos TE. O pior caso para CLIENTE_1 consome quase 10 vezes mais tempo que o pior caso na base CLIENTE_2. A melhor abordagem na primeira base ainda é 33% mais lenta que as melhores abordagens aplicadas na segunda base.

Os gráficos anteriores em representação *box plot* mostraram o desempenho de cada abordagem nas métricas específicas. No entanto, não existe uma melhor abordagem para todas as métricas.

IV) Análise Multi-Objetiva das Métricas MO

Para tratar as comparações de maneira mais justa foi concebida uma interpretação MO das próprias métricas utilizadas. A Figura 5.16 (CLIENTE_1) e a Figura 5.18 (CLIENTE_2) trazem todas as simulações ilustradas em espaços objetivos que confrontam o hipervolume (qualidade) e o espalhamento (diversidade) das soluções encontradas. A Figura 5.17 (CLIENTE_1) e a Figura 5.19 ilustram as curvas de nível dos tempos de execução destas simulações sobre estes mesmos espaços objetivos. Nestas figuras percebem-se ainda curvas denominadas de Pareto utópico que são representações do melhor *trade-off* possível entre as métricas HV e GS.

Apesar de existirem análises de três métricas, o *trade-off* apresentado entre elas pode ser facilmente reduzido em hipervolume *versus* espalhamento. Note-se que a região mais próxima ao Pareto utópico e, portanto onde estão as melhores soluções, também é a região que executa as simulações num menor espaço de tempo. Esta é uma observação importante, pois para o problema proposto o TE não é um objetivo conflitante com os demais.

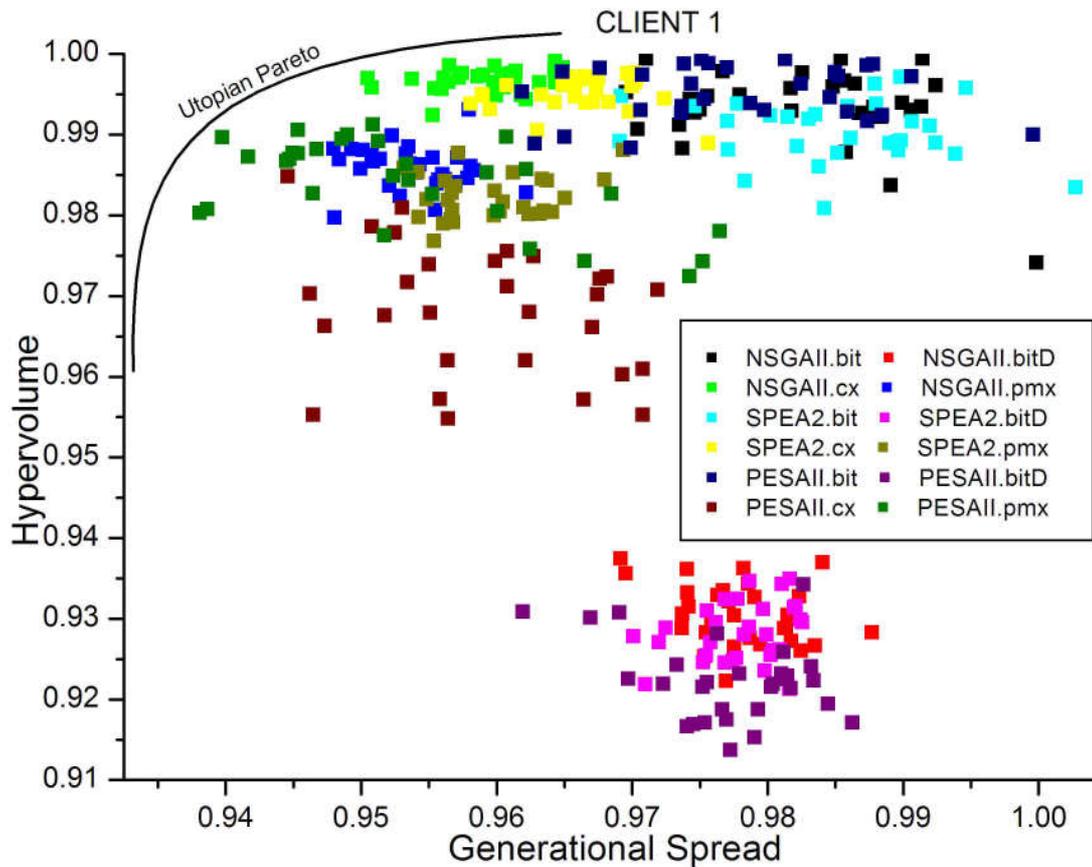


Figura 5.16. Espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_1.

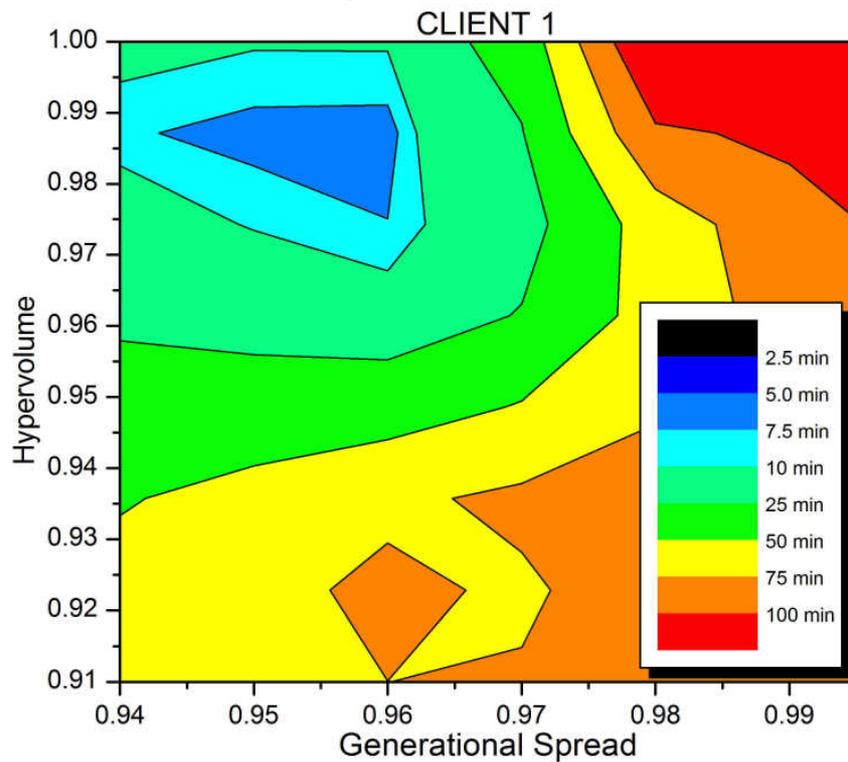


Figura 5.17. Curvas de nível de TE no espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_1.

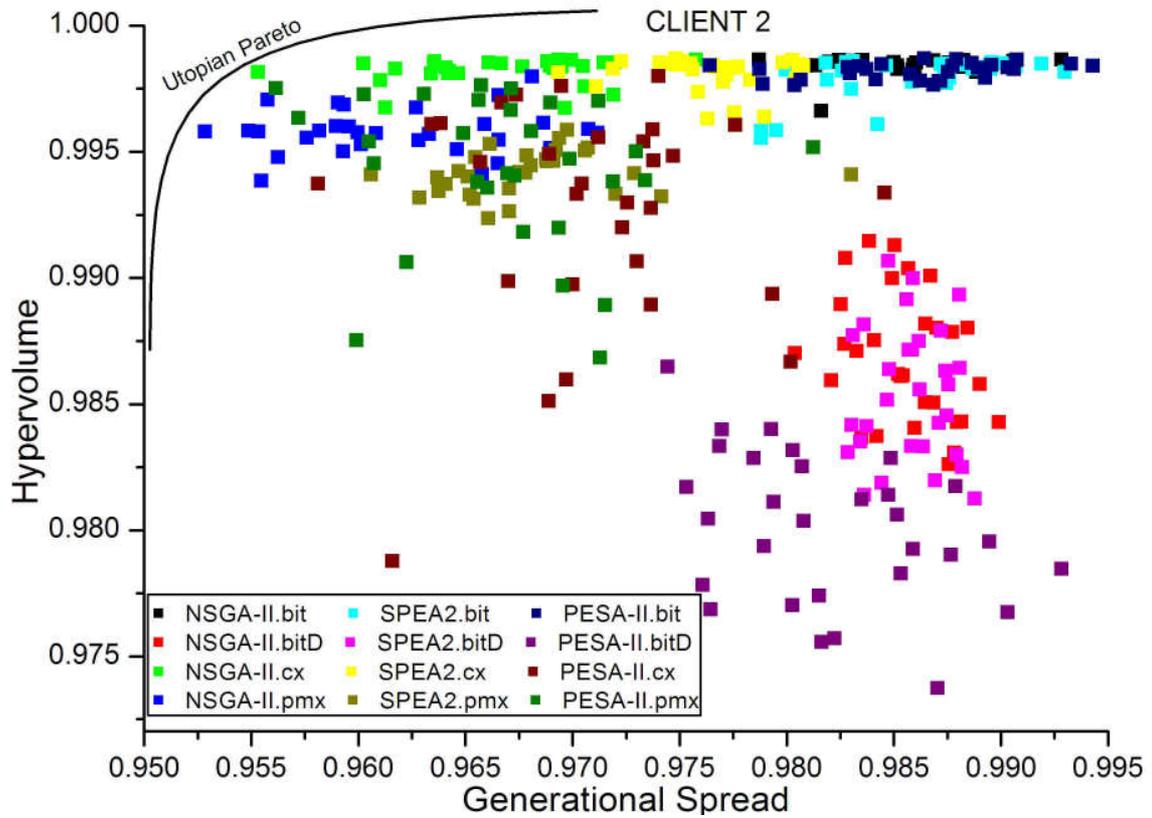


Figura 5.18. Espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_2.

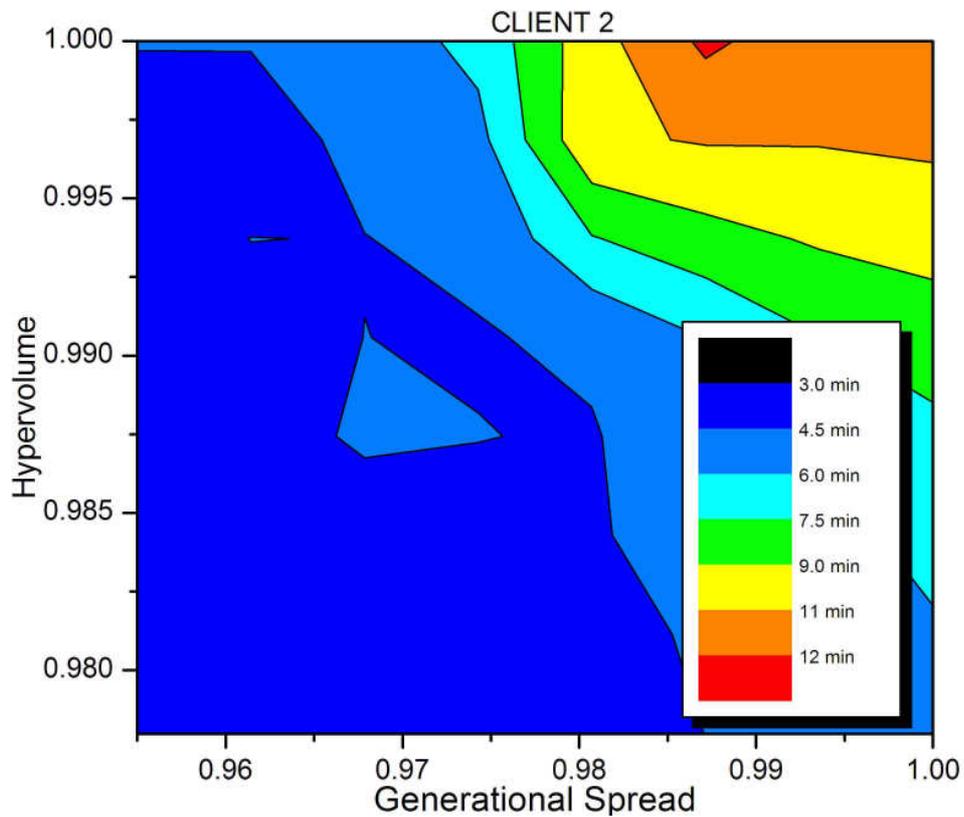


Figura 5.19. Curvas de nível de TE no espaço objetivo com HV e GS formado pelas amostras das 30 simulações de cada abordagem em CLIENTE_2.

5.3.3 Resultados da Fase Três

Com a análise exploratória a cerca da melhor forma de representar o problema, os melhores operadores genéticos e sobre quais algoritmos evolucionários se adaptam melhor na resolução do problema da colheita da cana-de-açúcar, inicia-se a última fase. Na seção 4.3.3, definiu-se que os objetivos desta fase seriam: (i) identificar a relevância das heurísticas utilizadas nos dias de colheita e (ii) desenvolver uma ferramenta que melhorasse essas decisões.

Nesta seção está descrito como os experimentos foram realizados, incluindo a configuração paramétrica, as métricas de desempenho e as abordagens utilizadas. Os dois níveis de aplicação de EMOO são detalhados.

O problema da colheita da cana-de-açúcar, como definido previamente, é formulado com quatro objetivos. Como um problema real ele pode ser avaliado a partir do atendimento destes objetivos (*i.e.* sua satisfação conjunta) em cenários que consideram (*i.e.* emulam) as preferências de um tomador de decisão, exatamente como na *Fase Um*. Já como um problema multi-objetivo ele pode ser avaliado a partir de métricas tradicionais, como a métricas apresentadas na *Fase Dois*: hipervolume [Zitzler1998], *generalized spread* [Zhou2006], dentre outras. Nesta fase, o problema da colheita foi avaliado de ambas as formas.

Todos os experimentos foram realizados com a base de dados real CLIENTE_2 que é composta de 590 lotes. O cenário geral de aplicação definiu que todos os lotes deveriam ser colhidos em um período de 6 meses. Assim, esta atividade foi dividida em 12 quinzenas (*i.e.* replicações do segundo nível de EMOO).

A. Configuração dos Parâmetros

Desde que há dois níveis de aplicação, duas configurações de parâmetros foram definidas e utilizadas para cada um dos algoritmos. Em ambos os casos, os ajustes dos parâmetros foram realizados experimentalmente.

1) Configuração do Primeiro Nível

O algoritmo NSGA-II foi usado para combinar seis heurísticas diferentes, uma para cada mês em todo o período de colheita. Estas heurísticas serão utilizadas em cada replicação da aplicação de segundo nível. Observe que a mesma heurística é aplicada em duas quinzenas (uma por mês) devido às similaridades dos indicadores agrônômicos – a predição dos indicadores não apresenta grandes modificações dentro de um curto período. O algoritmo utilizou uma representação real com mapeamento entre os números e as heurísticas, torneio binário,

cruzamento SBX e mutação polinomial [Deb2001]. A Tabela 5.9 mostra a configuração completa para o algoritmo do 1º nível.

Tabela 5.9. Configurações para o primeiro e o segundo nível de aplicação EMOO.

Parâmetros	1º nível	2º nível
Tamanho da população	30	200
Número de avaliações	1000	2000
Taxa de cruzamento	0.9	0.9
Taxa de mutação	1 / 6	1 / 590
Tonelagem desejada	-	50000
Tamanho do arquivo ¹²	-	200
Bi-section ¹²	-	5
Índice de distribuição ¹³	20	-

II) Configuração do Segundo Nível

Este nível é responsável pela principal otimização do problema – a colheita de fato. Os experimentos realizados na *Fase Dois* confrontando diferentes representações (*e.g.* binária, binária com decodificadores e simbólicas), operadores (*e.g.* cruzamentos de um-ponto, CX e PMX; mutação bit-flip e *swap*) e algoritmos (*e.g.* NSGA-II, SPEA2 e PESA-II) sugerem que a representação simbólica, com mutação *swap* e com os pares ‘algoritmo & cruzamento’ tais quais ‘NSGA-II & CX’ ou ‘PESA-II & PMX’ podem ser mais eficientes nas sugestões de colheita. A Tabela 5.9 mostra a configuração completa para o algoritmo do 2º nível.

B. Resultados

Para realizar os experimentos foram criadas 15 diferentes abordagens relacionadas com o conjunto de heurísticas criado na seção 4.3.3-C. A Tabela 5.10 descreve todas essas abordagens. Note que para ser equivalente e comparável com o trabalho desenvolvido na *Fase Um* [Pacheco2007], as abordagens de 1..5 na Tabela 5.10 não utilizam o primeiro nível de aplicação EMOO. Desta maneira, todas elas aplicaram a mesma heurística em todos os passos da colheita.

I) Avaliação Multi-objetiva

Esta avaliação só pode ser realizada nas abordagens que utilizam o primeiro nível de EMOO, pois, apresentam conjuntos de aproximações como saída de suas otimizações. Assim, as abordagens de 6..15 da Tabela 5.10 foram confrontadas considerando as métricas hipervolume e *generalized spread*.

¹² Aplicável apenas ao algoritmo PESA-II.

¹³ Aplicável aos operadores SBX e polinomial.

Tabela 5.10. Abordagens utilizadas nos experimentos da *Fase Três*.

Id	Abordagem	Descrição
1	<i>DISTANCIA--</i>	Aplica a heurística <i>REDUZIR_CUSTO</i> em todos os passos
2	<i>PCC++</i>	Aplica a heurística <i>AUMENTAR_PCC</i> em todos os passos
3	<i>ATR++</i>	Aplica a heurística <i>AUMENTAR_ATR</i> em todos os passos
4	<i>FIBRA++</i>	Aplica a heurística <i>AUMENTAR_FIBRA</i> em todos os passos
5	<i>EQUILIBRA</i>	Aplica a heurística Φ (28) em todos os passos
6	<i>3cpPESA</i>	Combina as heurísticas <i>crisp</i> de 1..3 com PESA-II
7	<i>3cpNSGA</i>	Combina as heurísticas <i>crisp</i> de 1..3 com NSGA-II
8	<i>4cpPESA</i>	Combina as heurísticas <i>crisp</i> de 1..4 com PESA-II
9	<i>4cpNSGA</i>	Combina as heurísticas <i>crisp</i> de 1..4 com NSGA-II
10	<i>5cpPESA</i>	Combina as heurísticas <i>crisp</i> de 1..5 com PESA-II
11	<i>5cpNSGA</i>	Combina as heurísticas <i>crisp</i> de 1..5 com NSGA-II
12	<i>fpPESA</i>	Combina as 6 heurísticas fuzzy com PESA-II
13	<i>fpNSGA</i>	Combina as 6 heurísticas fuzzy com NSGA-II
14	<i>allPESA</i>	Combina as 11 heurísticas com PESA-II
15	<i>allNSGA</i>	Combina as 11 heurísticas com NSGA-II

Na Figura 5.20, as abordagens simuladas são ilustradas confrontando as métricas HV e GS. Além disso, é determinando até que região do espaço de busca foi explorado por cada abordagem. O Pareto formado por elas é composto apenas por aproximações realizadas pelo algoritmo PESA-II e sem qualquer heurística fuzzy. Neste sentido, através da avaliação multi-objetiva, as abordagens que mais se aproximaram do Pareto ideal e conseguiram manter uma maior diversidade de soluções foram: *3cpPESA*, *4cpPESA* e *5cpPESA*. Ou seja, representam de maneira mais justa os *trade-offs* existentes entre os objetivos.

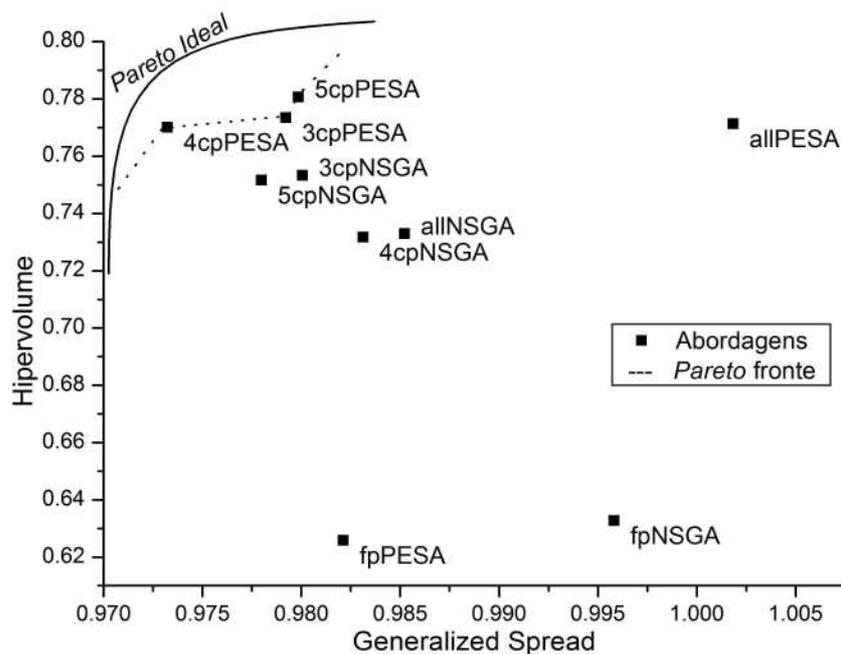


Figura 5.20. Comparação multi-objetiva de HV x GS das abordagens simuladas.

II) Avaliação de Problema Real

Na investigação desenvolvida na *Fase Um*, a avaliação foi realizada a partir da escolha de um objetivo específico, ou seja, apenas uma heurística foi aplicada durante toda a colheita. No caso, definiu-se o aumento da produção de PCC – calculada através da soma das produções diárias de colheita. Para evitar afirmações tendenciosas e possíveis generalizações não verdadeiras, foram concebidos cinco cenários hipotéticos diferentes para que as comparações fossem conclusivas. Quatro destes cenários representam os objetivos específicos e o último equilibra os ganhos e as perdas de todos os objetivos juntos.

As melhores cinco abordagens para cada um dos cinco cenários hipotéticos criados são exibidas na Tabela 5.11. Observe que a exceção do cenário CUSTOS, as variações entre as abordagens apresentam diferenças de porcentagens pequenas. Entretanto, devido às grandes escalas relacionadas a estes objetivos, o que é uma realidade na atividade sucro-alcooleira, estas diferenças mesmo que pequenas não devem ser ignoradas. Observe-se a Figura 5.21 que mostra o *trade-off* formado pelas soluções encontradas pela abordagem *5cpPESA*. A primeira vista, pode-se pensar que os objetivos agrônômicos são não-conflitantes. No entanto, a Figura 5.22 apresenta uma análise mais minuciosa (*zoom in*) que expõe os reais conflitos existentes.

Tabela 5.11. Classificação das cinco melhores abordagens entre as 15 pré-definidas e aplicadas em cada um dos cenários hipotéticos.

Cenário	Classificação				
	1º	2º	3º	4º	5º
PCC	<i>3cpNSGA</i> 100%	<i>5cpNSGA</i> 99.92%	<i>fpNSGA</i> 99.87%	<i>5cpPESA</i> 99.86%	<i>fpPESA</i> 99.85%
FIBRA	<i>5cpNSGA</i> 100%	<i>fpNSGA</i> 99.97%	<i>3cpNSGA</i> 99.96%	<i>4cpNSGA</i> 99.94%	<i>allPESA</i> 99.93%
ATR	<i>3cpNSGA</i> 100%	<i>4cpPESA</i> 99.98%	<i>allNSGA</i> 99.98%	<i>4cpNSGA</i> 99.96%	<i>3cpPESA</i> 99.93%
CUSTOS	<i>5cpPESA</i> 0.00%	<i>3cpPESA</i> 2.10%	<i>allPESA</i> 2.58%	<i>4cpPESA</i> 2.83%	<i>3cpNSGA</i> 8.17%
Equilíbrio	<i>5cpPESA</i> 2.981	<i>allPESA</i> 2.958	<i>3cpPESA</i> 2.954	<i>4cpPESA</i> 2.949	<i>5cpNSGA</i> 2.896

Para realizar uma interpretação mais clara destas comparações, definiu-se uma competição com o seguinte critério de pontuação:

- A melhor abordagem em determinado cenário recebe cinco pontos: $\rho_1^j = 5$, onde j representa os vários cenários;

- A partir da segunda colocada, em cada cenário, os pontos são determinados por: $\rho_{i-1}^j - 1$, onde i a colocação da abordagem no respectivo cenário;
- Assim, pontua-se apenas as cinco primeiras colocadas em cada cenário hipotético.

De acordo com essa pontuação, uma classificação entre as abordagens pode ser estabelecida, como mostra a Tabela 5.12. Assim, considerando possíveis situações do real problema da colheita da cana-de-açúcar, *3cpNSGA* é a abordagem que retorna melhores resultados para todos os cenários considerados, seguida de perto pelas abordagens *5cpPESA* e *5cpNSGA*.

Tabela 5.12. Classificação das abordagens de acordo com o critério de pontuação definido para os cenários hipotéticos.

Posição	Abordagem	Pontos
1	<i>3cpNSGA</i>	14
2	<i>5cpPESA</i>	12
3	<i>5cpNSGA</i>	10
4	<i>4cpPESA</i>	8
4	<i>3cpPESA</i>	8
4	<i>allPESA</i>	8
5	<i>fpNSGA</i>	7
6	<i>4cpNSGA</i>	4
7	<i>allNSGA</i>	3
8	<i>fpPESA</i>	1

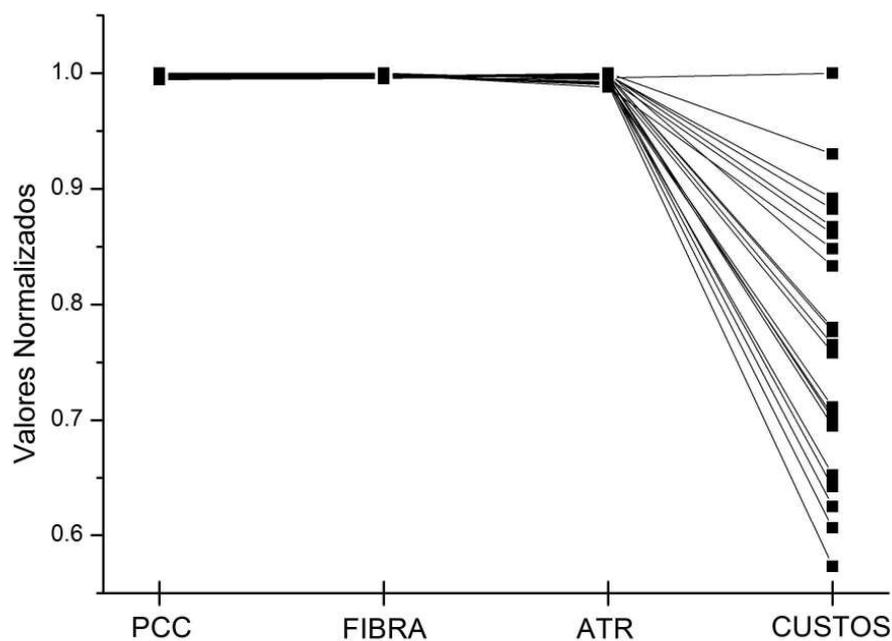


Figura 5.21. Trade-offs com objetivos agrônômicos e logístico obtido com a abordagem *5cpPESA*.

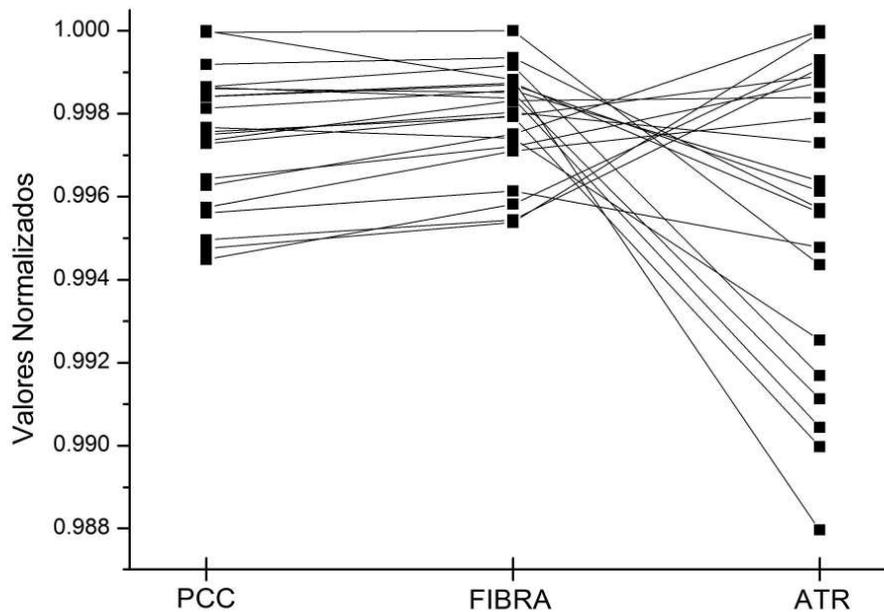


Figura 5.22. *Trade-offs* com apenas os objetivos agrônômicos obtidos com a abordagem *5cpPESA*.

5.3.4 Resumo dos Resultados

Ao final destas três fases, toda a proposta é comprovada através de experimentos. Primeiramente, os experimentos confrontando algoritmos clássicos e multi-objetivos, que favoreceu estes segundos. Na sequência, um estudo bastante minucioso sobre a representação do problema, assim como no desempenho dos algoritmos e operadores. Finalmente, na última fase, explorou-se os impactos causados pela interação – decisão de colheita – do tomador de decisão e o sistema. Os experimentos revelaram como o uso, em segundo nível, de EMOO pode beneficiar os resultados finais da colheita da cana-de-açúcar.

Capítulo 6

Conclusões e Trabalhos Futuros

Este trabalho cria um sistema inteligente híbrido (do inglês, *hybrid intelligent system*, HIS) multi-objetivo formado essencialmente por técnicas neurocomputacionais [Haykin1994] em conjunto com técnicas evolucionárias [Eiben2003] capaz de prever os indicadores agrônômicos da cana-de-açúcar, combinar com os dados logísticos e gerar ótimas sugestões de planejamento de colheita. Os experimentos deste trabalho foram realizados com três bases de dados reais de usinas brasileiras. Os resultados obtidos revelam como a produtividade final da colheita pode ser aumentada e como os gerentes agrícolas podem ser beneficiados com a ferramenta.

6.1 Contribuições

Os objetivos traçados para este trabalho de pesquisa foram desenvolvidos e alcançados em fases. O sistema inteligente híbrido desenvolvido, composto por módulos preditivos, otimizadores e decisores, foi concebido ao longo destas fases. Assim, as conclusões têm mais relevância e clareza quando apresentadas divididas nestas mesmas fases. Por outro lado, para que estas fases pudessem ocorrer, toda uma modelagem neural (módulo preditivo) foi desenvolvida a priori para cada uma das bases de dados utilizadas; as quais, por sua vez, precisaram de um pré-processamento. Os novos resultados de predição obtidos neste trabalho se mantêm compatíveis aos encontrados para as demais bases de dados utilizadas anteriormente.

6.1.1 Contribuições Fase Um

Na maioria das usinas as decisões de colheita são tomadas, de maneira empírica ou heurística. A adoção de técnicas inteligentes ajustadas para apoiar os tomadores de decisão nesta atividade

pôde aumentar a produtividade das safras. Esta fase mostrou que a aplicação de algoritmos evolucionários com uma formulação multi-objetivo pode ser ainda mais benéfica ao processo decisório quando comparada com a aplicação de métodos multi-objetivos clássicos.

O algoritmo NSGA-II, da maneira que foi modelado, agrega muito mais propriedade e justiça às decisões gerenciais. A avaliação realizada considerando todo o período de colheita representa uma contribuição real para o problema. Enquanto que em trabalhos anteriores, com a avaliação de dias isolados de colheita, apenas revelava este potencial.

A abordagem *b*, que considerou a restrição de tonelagem do problema como um novo objetivo, demonstrou que esta linha de pensamento é satisfatória e consegue obter resultados ainda melhores. O uso de uma função de penalidade, como na abordagem *a*, pode causar os mesmos problemas enfrentados pelos métodos MO clássicos. Isto é, o uso prematuro de informações de mais alto nível, a partir da definição de uma função de penalidade, não permite mensurar os reais impactos destas decisões no conjunto de aproximação. Ademais, o conhecimento a priori a cerca destas preferências, muitas vezes não é suficiente para determinar funções de penalidade (*i.e.* pesos entre os objetivos) eficientes.

6.1.2 Contribuições Fase Dois

Nesta fase o problema da colheita da cana-de-açúcar foi instanciado como um problema de múltiplas mochilas (MKP). Incorporou-se ainda dados logísticos na resolução do problema, o que trouxe mais realismo às simulações. Ao comparar estes dois problemas, pode-se explorar diferentes formas de representação, diferentes operadores genéticos e diferentes algoritmos evolucionários multi-objetivo que são o estado-da-arte. Para realizar os experimentos e a validação da proposta, formulamos dois problemas de colheita distintos para duas bases de dados de usinas brasileiras. Os problemas foram projetados para investigar algumas características, tais como:

- Múltiplos objetivos na colheita da cana-de-açúcar: o problema formulado contou com 4 objetivos e 2 restrições;
- Escalabilidade: utilizaram-se duas bases de dados representativas envolvidas na otimização, com 937 e 590 lotes cada uma;
- Restritividade (*restrictiveness*): tonelagens desejadas (capacidade da mochila) bastante restritivas, 10% e 30% para as bases CLIENTE_1 e CLIENTE_2, respectivamente;

- Avaliações justas: comparações de resultados consideraram a qualidade (hipervolume), a diversidade no Pareto *front* (*generalized spread*) e o desempenho (tempo de execução) de todos os conjuntos de aproximação.

Os resultados sugerem que a representação simbólica baseada em ordem (*order-based*) com os operadores de cruzamento CX ou PMX é recomendada para os algoritmos NSGA-II e SPEA2; para o algoritmo PESA-II apenas o operador PMX é recomendado junto à representação simbólica. A representação binária *bit* alcançou bons resultados para a métrica hipervolume, entretanto, seu desempenho foi bastante aquém com respeito à diversidade de soluções e ao tempo de execução das simulações. Na medida em que a complexidade do problema aumenta, o uso de decodificadores com representação binária (*bitD*) perdeu a efetividade.

6.1.3 Contribuições Fase Três

Nesta fase investigou-se como as preferências diárias afetavam a produtividade final no problema da colheita da cana-de-açúcar. A partir de uma meta aplicação de algoritmos evolucionários multi-objetivo para combinar diferentes heurísticas durante o ciclo da colheita, conseguiu-se aumentar a produtividade para todos os cenários hipotéticos criados.

Apesar das pequenas variações entre os objetivos parcialmente camuflar os *trade-offs* existentes entre eles, uma análise cuidadosa mostrou ser importante não desprezar essa multi-objetividade. As pequenas diferenças nas saídas encontradas pelas diversas abordagens podem levar um leitor menos atento a crer na irrelevância de uma perante a outra, entretanto, devido aos volumes financeiros negociados na indústria canavieira, essas diferenças não podem ser ignoradas.

Finalmente, entre os algoritmos testados (sugeridos após a *Fase Dois*), não ficou claro se o algoritmo PESA-II é de fato melhor que o algoritmo NSGA-II, já que ambos alcançam bons resultados. Em adição, o uso de lógica fuzzy na definição das heurísticas aplicadas não conseguiu melhorar os resultados finais, mas sua aplicação é um grande facilitador na interação homem-sistema.

6.2 Discussão

Acreditamos que a utilização deste sistema acoplado a uma interface gráfica amigável em ambientes reais da indústria canavieira pode determinar uma melhor produtividade no cultivo da cana-de-açúcar, pois os resultados obtidos demonstram que sua aplicação consegue, ao mesmo

tempo, melhorar a qualidade da matéria-prima e reduzir os custos logísticos. Certamente, esta melhoria pode aumentar a rentabilidade das usinas e facilitar as decisões dos gestores. Adicionalmente, isto ainda pode reduzir o preços finais dos produtos industrializados decorrentes desta matéria prima agrícola.

A concepção em fases do sistema permitiu um melhor amadurecimento do modelo o que resultou em bastante coesão. A evolução da proposta permite o desenvolvimento de extensões do modelo produzido neste mestrado, para gerar curvas de maturação das variedades de cana-de-açúcar desejadas em vários cenários diferentes. Isto possibilita ao agente decisor conhecer o momento exato para a colheita de qualquer variedade cultivada em sua unidade, inclusive considerando todos os efeitos extra-temporâneos, tais como seca, tratos culturais, pragas etc. Permite ainda a inclusão de diversos outros fatores logísticos, tornando o sistema tão complexo quanto o necessário para que as contribuições aqui descritas sejam refletidas em melhores rentabilidades. Já a formulação multi-objetivo possibilita que as decisões sejam tomadas de um modo mais justo, com um tratamento indiferenciado entre os objetivos. Por outro lado, havendo diferença nas preferências entre esses objetivos, estas podem ser incorporadas pelo sistema de modo que este se torna uma extensão do próprio decisor.

A solução ora apresentada, desde que treinada com dados histórico de produção, pode ser aplicada às mais diversas regiões agrícolas do país. Isto oferece uma flexibilidade interessante e necessária para unidades de produção instaladas em mais de uma região. A pesquisa científica aplicada dentro deste setor econômico contribui ainda mais para o desenvolvimento do agronegócio brasileiro e ainda firma novas possibilidades de estudos e aplicações. Assim sendo, a pesquisa científica poderá, cada vez mais, gerar ciência em cima de problemas reais, desenvolvendo nossa região com aplicação direta de novos conhecimentos.

6.3 Trabalhos Futuros

Como trabalhos futuros, algumas linhas de pesquisa podem ser definidas:

- Aprimorar o módulo preditivo e melhorar os valores de predição a partir da mudança dos paradigmas utilizados. Tentar incorporar dados meteorológicos na modelagem neural ou traçar diferentes regiões nas plantações de cana-de-açúcar. Pode-se ainda realizar novas buscas paramétricas ou até mesmo novas técnicas para predição. Possibilidade de aplicar algoritmos genéticos para descobrir a topologia e os parâmetros destas RNA;

- Modificar a maneira como o sistema define a colheita, ou seja, ao invés de escolher o melhor conjunto de lotes para cada dia (visão vertical), tentar descobrir o melhor momento de colheita de cada lote (visão horizontal);
- Investigar a articulação progressiva de preferências, em especial nos algoritmos de segundo nível (*Fase Três*), o que pode reduzir significativamente o tempo gasto por essas meta-aplicações;
- Investigar como a combinação de técnicas inteligentes pode contribuir ainda mais em outras atividades da cultura canavieira (que não a colheita), por exemplo: plantio, adubação, fertilização, irrigação, armazenamento, entre outros;
- Desenvolver um projeto piloto, que aplique estes conceitos não apenas em dados reais, mas em ambientes industriais reais;
- Finalmente, extrapolar as idéias propostas neste mestrado e aplicadas no cultivo da cana-de-açúcar em outros domínios agrícolas, tais como: soja, uva, algodão, entre outros.

Bibliografia

- [Abbod2007] M. F. Abbod, J. W.F. Catto, D. A. Linkens and F. C. Hamdy, “Application of artificial intelligence to the management of urological cancer”, *The Journal of Urology*, Vol. 178, Issue 4, Oct. 2007, pp. 1150-1156.
- [Agroclubes2008] Clube do Açúcar e do Alcool, "Ficha técnica". Acesso em Junho de 2008. Disponível em http://www.agroclubes.com.br/ficha_tecnica/fichas.asp?ficha=884&codigo_produto=884.
- [Alba2007] E. Alba, “Cellular genetic algorithms”, Tutorial in Seventh International Conference on Intelligent Systems Design and Applications (ISDA), Rio de Janeiro, Brazil, Oct. 2007.
- [Alencar2006] C. E. R. Alencar, R. F. Corrêa, F. B. Lima Neto, “Ferramenta para suporte à decisão de frentes de corte de cana-de-açúcar usando algoritmos genéticos”. Anais do WCI 2006 - I Workshop on Computational Intelligence (Evento adjunto ao International Joint Conference 2006: IBERAMIA / SBIA / SBRN), Ribeirão Preto, Brasil, 2006.
- [Alexander1973] A. G. Alexander, “Sugarcane physiology: a comprehensive study of the Saccharum source-to-sink system”, Elsevier, Amsterdam, 1973.
- [Aranha1987] C. Aranha, C. A. Yahn, “Botânica da cana-de-açúcar”, em *Cana-de-Açúcar: Cultivo e Utilização*, Capítulo I, por Sérgio B. Paranhos, Coordenador. Fundação Cargil, Campinas, SP, Brasil, 1987.
- [BastosFilho2008] C. J. A. Bastos Filho, F. B. de Lima Neto, A. J. C. C. Lins, A. I. S. Nascimento, M. P. Lima, "A Novel Search Algorithm based on Fish School Behavior". In: IEEE International Conference on Systems, Man, and Cybernetics - SMC2008, Outubro, 2008, Singapore.
- [Burkard1994] R. E. Burkard, U. Pferschy, “The inverse-parametric knapsack problem”, Technische Universität Graz, Report 94/288, Austria, 1994.
- [Caldas2007] B. J. de B. Caldas, M. R. S. Pita, F. B. de Lima Neto, "How to obtain appropriate executive decisions using artificial immunologic systems". In: 6th International Conference on Artificial Imune Systems (ICARIS), Santos – Brasil, 2007.
- [Carpenter1988] G. A. Carpenter, S. Grossberg, "The ART of adaptive pattern recognition by a self-organizing neural network", *Computer*, vol.21, no.3, pp.77-88, 1988.

- [Carvalho2008] D. F. Carvalho, C. J. A. Bastos Filho, "Clan particle swarm optimization". In: 2008 IEEE Congress on Evolutionary Computation (IEEE CEC 2008) within 2008 IEEE World Congress on Computational Intelligence (WCCI 2008), Hong Kong, China.
- [Coello1999] C. A. C. Coello, "A comprehensive survey of evolutionary-based multiobjective optimization". Knowledge and Information Systems, 1999.
- [Coello2000] C. A. C. Coello, "Handling preferences in evolutionary multiobjective optimization: a survey", In Proceedings of Evolutionary Computation, vol.1, pp.30-37, 2000.
- [Coello2007a] C. A. C. Coello, "Constraint-handling techniques used with evolutionary algorithms", Tutorial, In 2007 IEEE Congress on Evolutionary Computation (CEC'2007), 2007.
- [Coello2007b] C. A. C. Coello, "Metaheuristics for multiobjective optimization". Tutorial In: IEEE Symposium Series on Computational Intelligence, vol.1, Hawaii, 2007. CD-ROM.
- [Cohon1985] J. L. Cohon, "Multicriteria programming: brief review and application". In J. S. Gero (Ed.), Design Optimization, pp. 163–191. Academic Press, Orlando, Florida, 1985.
- [Colombo2005] G. Colombo and C. Mumford, "Comparing Algorithms, Representations and Operators for the Multi-Objective Knapsack Problem", in 2005 IEEE Congress on Evolutionary Computation (CEC'2005), pp. 1268--1275, Vol. 2, IEEE Service Center, Edinburgh, Scotland, September 2005.
- [Copersucar1988] Cooperativa dos produtores de cana, açúcar e álcool do Estado de São Paulo (Copersucar), "Recomendação de adubação para a cultura da cana-de-açúcar". Piracicaba – SP: Centro de Tecnologia Copersucar, 1988. 7 p.
- [Copersucar2008] Academia do Açúcar e do Alcool, "Cana-de-açúcar", Copersucar. Acesso em Maio de 2008. Disponível em http://www.copersucar.com.br/institucional/por/academia/cana_acucar.asp
- [Corne2000] D. W. Corne, J. D. Knowles, M. J. Oates, "The Pareto envelope-based selection algorithm for multiobjective optimization", In Marc Schoenauer, Kalyanmoy Deb, Günter Rudolph, Xin Yao, Evelyne Lutten, J. J. Merelo and Hans-Paul Schwefel (editors), Proceedings of the Parallel Problem Solving from Nature VI Conference, pp. 839-848. Springer, 2000.
- [Corne2001] D. W. Corne, N. R. Jerram, J. D. Knowles, M. J. Oates, "PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization", Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), pp. 283--290, Morgan Kaufmann Publishers, San Francisco, California, 2001.
- [Cunha1997] A. G. Cunha, P. Oliviera, J. Covas, "Use of genetic algorithms in multicriteria optimization to solve industrial problems". In T. Bäck (Ed.), Proceedings of the Seventh International Conference on Genetic Algorithms, San Francisco, California, pp. 682–688. Morgan Kaufmann, 1997.

- [Dantzig1957] G. B. Dantzig. “Discrete variable extremum problems”, *Operations Research*, 5, pp. 266-277, 1957.
- [Darwin1989] C. Darwin, “A origem das espécies e a seleção natural”, Hemus editora, 1989.
- [Deb1989] K. Deb, “Genetic algorithms in multi-modal function optimization”, Master’s Thesis, Tuscaloosa, AL: University of Alabama, 1989.
- [Deb1995] K. Deb, “Optimization for engineering design: algorithms and examples”, Prentice-Hall, New Delhi, 1995.
- [Deb1999] K. Deb, “Evolutionary algorithms for multi-criterion optimization in engineering design”. In *Proceedings of Evolutionary Algorithms in Engineering and Computer Science (EUROGEN’99)*, 1999.
- [Deb2000a] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, “A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II”, KanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.
- [Deb2000b] K. Deb, J. Horn, “Introduction to the special issue: multicriterion optimization”, *Evolutionary Computation Journal* 8 (2), iii-iv, 2000.
- [Deb2001] K. Deb. “Multi-objective optimization using evolutionary algorithms”, John Wiley & Sons, UK, 2001.
- [Deb2002] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II”, *Evolutionary Computation*, IEEE Transactions on, vol.6, no.2, pp.182-197, Apr 2002.
- [DeJong1975] K. A. DeJong, “An analysis of the behavior of a class of genetic adaptive systems”. Ph. D. Thesis, Ann Arbor, MI: University of Michigan, 1975.
- [Dudzinski1987] K. Dudzinski, S. Walukiewicz, “Exact methods for the knapsack problem and its generalizations”, *European Journal of Operational Research*, 28, pp. 3-21, 1987.
- [Duque2007] W. Ocampo-Duque, M. Schuhmacher and J. L. Domingo, “A neural-fuzzy approach to classify the ecological status in surface waters”, *Environmental Pollution*, Vol. 148 (2), pp 634-641, 2007.
- [Eiben2003] A. E. Eiben, J. E. Smith, “Introduction to evolutionary computing”. Springer, New York, 2003.
- [FAO2007] Food and Agricultural Organization of United Nations, “Major Food and Agricultural Commodities and Producers”, The Statistics Division, Economic and Social Department, available on <http://www.fao.org/es/ess/top/commodity.html?lang=en&item=156&year=2005>, accessed in May 2007.
- [Fayard1994] D. Fayard, G. Plateau, “An exact algorithm for the 0-1 collapsing knapsack problem”, *Discrete Applied Mathematics*, 49, pp. 175-187, 1994.

- [Figueira2004] J. Figueira *et al.*, “Multiple criteria decision analysis: state of the art surveys”, Springer, New York, 2004.
- [Figueira2005] S. R. Figueira, “Os programas de álcool como combustíveis nos EUA, no Japão e na União Européia e as possibilidades de exportação do Brasil”. Tese (Doutorado em Ciências) – Escola Superior de Agricultura “Luiz de Queiroz”, Universidade de São Paulo, Piracicaba, 245p. 2005.
- [Fi-Liu2004] L. M. Fu, C. S. Fu-Liu, “Multi-class cancer subtype classification based on gene expression signatures with reliability analysis”, *FEBS Letters*, Vol. 561 (1-3), pp. 186-190, 2004.
- [Fleming2005] P. J. Fleming, R. C. Purshouse, R. J. Lygoe, “Many-objective optimization: an engineering design perspective”, *Lecture Notes in Computer Science* 3410, pp. 14-32, 2005.
- [Fonseca1995] C. M. Fonseca, P. J. Fleming, “An overview of evolutionary algorithms in multi-objective optimization”, *Evolutionary Computation Journal* 3 (1), pp. 1-16, 1995.
- [Fonseca1996] C. M. Fonseca, P. J. Fleming, “On the performance assessment and comparison of stochastic multiobjective optimizers”. In *Proceedings of Parallel Problem Solving from Nature IV (PPSN-IV)*, pp. 584-593, 1996.
- [Fonseca1998a] C. M. Fonseca, P. J. Fleming, “Multiobjective optimization and multiple constraint handling with evolutionary algorithms - part I: a unified formulation”, *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, vol.28, no.1, p.26-37, Jan 1998.
- [Fonseca1998b] C. M. Fonseca, P. J. Fleming, “Multiobjective optimization and multiple constraint handling with evolutionary algorithms – part II: application example”. *IEEE Transactions on Systems, Man, and Cybernetics* 28 (1), 38–47, 1998.
- [Fonseca2005] C. M. Fonseca, J. D. Knowles, L. Thiele, E. Zitzler, “A tutorial on the performance assessment of stochastic multiobjective optimizers”, Presented at the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005).
- [Fonseca2007] C. M. Fonseca, “Preference articulation in evolutionary multiobjective optimization”, *HIS, Tutorial*, p. 4, 7th International Conference on Hybrid Intelligent Systems (HIS 2007), 2007.
- [Fortes2003] C. Fortes, “Discriminação varietal e estimativa de produtividade agroindustrial de cana-de-açúcar pelo sensor orbital ETM+/LANDSAT 7”. Dissertação (Mestrado em Agronomia), Escola Superior de Agricultura “Luiz de Queiroz”, Universidade de São Paulo, Piracicaba, 131p. 2003.
- [Fülöp2005] J. Fülöp, “Introduction to decision-making methods”, *BDEI-3 Workshop*, Washington, 2005.
- [Gilmore1963] P. C. Gilmore, R. E. Gomory, "A linear programming approach to the cutting stock problem - part II", *Operations Research: INFORMS*, Vol. 11, No. 6, pp. 863-888, 1963.

- [Goldberg1985] D. E. Goldberg and R. Lingle, "Alleles, loci and the TSP", Proceedings of an International Conference on Genetic Algorithms and Their Applications, Pittsburgh, PA, pp. 154–159, 1985.
- [Goldberg1989] D. E. Goldberg, "Genetic algorithms for search, optimization, and machine learning", Addison-Wesley, Reading, MA, 1989.
- [Goldberg1991] D. E. Goldberg, K. Deb, "A comparison of selection schemes in genetic algorithms". In Foundations of Genetic Algorithms 1 (FOGA-1), pp. 220 – 228, 1991.
- [Gordon1993a] V. S. Gordon, D. Whitley, "Serial and parallel genetic algorithms as function optimizers". In Proceeding of the Fifth International Conference on Genetic Algorithms. Morgan Kaufmann, pp. 177-183, 1993.
- [Gordon1993b] V. S. Gordon, A. P. W. Bohm, D. Whitley, "A note on the performance of genetic algorithms on zero-one knapsack problems". Technical report cs-93-108, Department of Computer Science, Colorado State University, USA, 1993.
- [Gore2006] A. Gore, "An inconvenient truth. The planetary emergency of global warming and what we can do about it", Rodale Books, New York - USA, 2006.
- [Gupta2007] K. K. Gupta, R. Gupta, "Adaptive shrinkage function optimization by differential evolution", International Conference on Advanced Computing and Communications (ADCOM 2007), pp.776-779, 2007.
- [Hajela1992] P. Hajela, C.-Y. Lin, "Genetic search strategies in multi-criterion optimal design", Structural Optimization 4 (2), pp.99-107, 1992.
- [Hansen1998] M. P. Hansen, A. Jaskiewicz, "Evaluating the quality of approximations to the non-dominated set", Technical Report IMM-REP-1998-7, Lyngby: Institute of Mathematical Modelling, Technical University of Denmark, 1998.
- [Haykin1994] S. Haykin, "Neural networks – a comprehensive foundation". Prentice-Hall International Editions, New Jersey, USA, 1994.
- [Hinterding1994] R. Hinterding, "Mapping, order-independent genes and the knapsack problem". In Proceedings of the First IEEE Conference on Evolutionary Computation, Orlando, Florida, IEEE Press, pp. 13-17, 1994.
- [Hinterding1998] R. Hinterding, Z. Michalewicz, "Your brains and my beauty: Parent matching for constrained optimisation". In IEEE International Conference on Evolutionary Computation. IEEE Press, 1998.
- [Hinterding1999] R. Hinterding, "Representation, constraint satisfaction and the knapsack problem", CEC99. Proceedings of the 1999 Congress on Evolutionary Computation, vol.2, no., pp.1286-1292 Vol. 2, 1999.
- [Holland1975] J. H. Holland, "Adaptation in natural and artificial systems". Ann Arbor, MI: MIT Press, 1975.

- [Horn1994] J. Horn, N. Nafploitis, D. Goldberg, “A niched Pareto genetic algorithm for multi-objective optimization”, In Proceedings of the First IEEE Conference on Evolutionary Computation, pp. 82-87, 1994.
- [Horn1997] J. Horn, “Multicriteria decision making”, In T. Bäck, D. B. Fogel, and Z. Michalewicz (Eds.), Handbook of Evolutionary Computation. Institute of Physics Publishing, Bristol (UK), 1997.
- [Hu2007] P. Jen-Hwa Hu, Chih-Ping Wei, Tsang-Hsiang Cheng and Jian-Xun Chen, “Predicting adequacy of vancomycin regimens: a learning-based classification approach to improving clinical decision making”, Decision Support Systems, Vol. 43 (4), Special Issue Clusters, pp. 1226-1241, 2007.
- [Huang2007] Shi-Ming Huang, Chih-Fong Tsai, D. C. Yen and Yin-Lin Cheng, “A hybrid financial analysis model for business failure prediction”, Expert Systems with Applications (in Press), Corrected Proof, Available online 15th August of 2007.
- [Hwang1979] C.-L. Hwang, A. S. M. Masud, “Multiple objectives decision making - methods and applications”, Springer, Berlin, 1979.
- [Ibaraki1987a] T. Ibaraki, “Enumerative approaches to combinatorial optimization – part 1”, Annals of Operations Research, 10, 1987.
- [Ibaraki1987b] T. Ibaraki, “Enumerative approaches to combinatorial optimization – part 2”, Annals of Operations Research, 11, 1987.
- [IBPS2008] Instituto Brasileiro de Produção Sustentável e Direito Ambiental (IBPS), “Cana-de-açúcar supera hidrelétrica na produção de energia”. Disponível em <http://www.ibps.com.br/index.asp?idnoticia=4236>. Acesso em Junho de 2008.
- [iLab2008] iLab Sistemas Especialistas. Disponível em <http://www.ilab.com.br/>. Acesso em junho de 2008.
- [Investing2007] Biodiesel and Ethanol Investing, “Ethanol fuel benefits”. Available on <http://www.biodieselinvesting.com/ethanol-fuel-benefits/>. Accessed in May 2007.
- [Ishibuchi1996] H. Ishibuchi, T. Murata, “Multi-objective genetic local search algorithm”. In Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC'96), Piscataway, NJ, pp. 119–124, 1996.
- [Jain1996] A. K. Jain, J. Mao, K. M. Mouhiuddin, “Artificial neural networks: a tutorial”. IEEE Computer, pp. 31-44, 1996.
- [Keating1999] B. A. Keating, M. J. Robertson, R. C. Muchow, N. I. Huth, “Modelling sugarcane production systems I. Development and performance of the sugarcane module”, Field Crops Research, Vol. 61, Issue 3, pp. 253-271, 1999.
- [Khor2007] M. Khor, “Alimentos baratos: o fim de uma era”, ONG Rede do Terceiro Mundo, Economia - Revista Mirada Global, 2007. Disponível em: http://www.miradaglobal.com/index.php?option=com_content&task=view&id=698&Itemid=9&lang=pt. Acesso em maio de 2008.

- [Khuri1994] S. Khuri, T. Bäck, J. Heitkötter, "The zero-one multiple knapsack problem and genetic algorithms". In Proceedings of the 1994 ACM Symposium of Applied Computation. ACM Press, 1994.
- [Knowles1999] J. Knowles, D. Corne, "Assessing the performance of the pareto archived evolution strategy". In A. S. Wu (Ed.), Proceedings of the 1999 Genetic and Evolutionary Computation Conference (GECCO'99), Workshop Program, pp. 123–124, 1999.
- [Knudson2007] M. J. Knudson, J. C. Austin, M. Wald, A. A. Makhlof, C. S. Niederberger, C. S. Cooper, "Computational model for predicting the chance of early resolution in children with vesicoureteral reflux", The Journal of Urology, Vol. 178 (4), pp. 1824-1827, 2007.
- [Kohonen1995] T. Kohonen, "Self organizing maps", Springer, Berlin, 1995.
- [Kolesar1967] P. J. Kolesar, "A branch and bound algorithm for the knapsack problem", Management Science, 13, pp. 723-735, 1967.
- [Kuhn1951] H. W. Kuhn, A. W. Tucker, "Nonlinear programming". In: Proceedings of Second Berkeley Symposium on Mathematical Statistics and Probability, J. Neyman, Ed. University of California at Berkeley, Berkeley, CA, pp.481-492, 1951.
- [Kursawe1990] F. Kursawe, "A variant of evolution strategies for vector optimization". In Parallel Problem Solving from Nature I (PPSN-I), pp.193-197, 1990.
- [Laudon2000] K. C. Laudon, J. P. Laudon, "Management information systems: organization and technology in the networked enterprise". Sexta edição, editora: Prentice-Hall, New Jersey, 2000.
- [Laumanns1998] M. Laumanns, G. Rudolph, H.-P. Schwefel, "A spatial predator-prey approach to multi-objective optimization: a preliminary study". In A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel (Eds.), Fifth International Conference on Parallel Problem Solving from Nature (PPSNV), Berlin, Germany, pp. 241–249. Springer, 1998.
- [LimaNeto1997] F. B. L. Neto, T. B. Ludermir. "Neural decision support for the harvesting of sugar-cane", In: IV Simpósio Brasileiro de Redes Neurais, 1997, Goiania. Anais do IV Simpósio Brasileiro de Redes Neurais, 1997.
- [LimaNeto1998] F. B. Lima Neto, "Suporte à decisão gerencial baseado em redes neurais artificiais – nDSS", Dissertação de Mestrado, Departamento de Informática, Universidade Federal de Pernambuco, Recife, Brasil, 1998.
- [LimaNeto2007] F. B. Lima Neto, F. R. S. Oliveira and D. F. Pacheco, "HIDS: Hybrid intelligent suite for decision support", Proceedings of Seventh International Conference on Intelligent Systems Design and Applications (ISDA), Rio de Janeiro, Brazil, Oct. 2007, pp. 90-95.
- [Lin2008] Wennan Lin, Yihua Li, Xingtao Xu, Maojun li, "Reactive power optimization in area power grid based on improved Tabu search algorithm", In Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies (DRPT 2008), pp.1472-1477, 2008.

- [Linkov2004] I. Linkov, et. al., “Multi-criteria decision analysis: a framework for structuring remedial decisions at the contaminated sites”, Springer, New York, 2004, pp. 15-54.
- [Lis1997] J. Lis, A. E. Eiben, “A multi-sexual genetic algorithm for multiobjective optimization”. In Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC’97), Piscataway, NJ, pp. 59–64, 1997.
- [Llewellyn2007] J. Llewellyn, “The business of climate change: challenges and opportunities”, Lehman Brothers, February 2007. Available in: http://www.lehman.com/press/pdf_2007/TheBusinessOfClimateChange.pdf. Accessed in May 2008.
- [Lourenço2008] L. Lourenço, “Conferência da FAO é marcada por embate sobre produção de alimentos e de biocombustíveis”, Agência Brasil, EBC – Empresa Brasil de Comunicação, 14 de Abril de 2008. Disponível em: <http://www.agenciabrasil.gov.br/noticias/2008/04/14/materia.2008-04-14.0433088353/view>. Acesso em maio de 2008.
- [Lucas1997] B. M. Lucas, “Information technology for management”. 6th ed. New York: McGraw-Hill, 1997.
- [Lucas2007] A. de A. Lucas, C. A. B. Schuler, “Analysis of NDVI/NOAA in sugar cane and Atlantic Forest in the north of Pernambuco state, Brazil”. Revista brasileira de engenharia agrícola ambiental, Campina Grande, v. 11, n. 6, 2007.
- [Lygoe2005] R. J. Lygoe, P. J. Fleming, “An Understanding of Fonseca & Fleming's preferability operator with respect to the decision making process in multi-objective optimization”, ACSE Research Report 880, University of Sheffield, Sheffield, UK, 2005.
- [Machado1987] E. C. Machado, “Fisiologia de produção de cana-de-açúcar”, Cap. em: S. B. Paranhos, Cana-de-açúcar: cultivo e utilização. Campinas, SP, Brasil: Fundação Cargil, pp. 56-87, 1987.
- [Machado2008] F. de B. P. Machado, “A história da cana-de-açúcar - da antiguidade aos dias atuais”. Resumo do texto *Brasil, a doce terra*. Disponível em <http://www.toneis.com.br/modules.php?name=News&file=article&sid=55>. Acesso em Junho de 2008.
- [Magalhães1987] A. C. N. Magalhães, “Ecofisiologia da cana-de-açúcar: aspectos do metabolismo do carbono da planta”. In: P. R. C. Castro, S. O. Ferreira, T. Yamada (Coord.), Ecofisiologia da Produção. Piracicaba: Associação Brasileira para pesquisa do Potássio e do Fósforo, pp. 113-118, 1987.
- [McCulloch1943] W. S. McCulloch, W. Pitts, “A logical calculus of the ideas immanent in nervous activity”. Bulletin of Mathematical Biophysics, 1943.
- [McNurlin2005] B. C. McNurlin, R. H. Sprague, "Information Management in Practice", 7th. Edition, Prentice Hall, 2005.
- [Michalewicz1994] Z. Michalewicz and J. Arabas, Genetic Algorithms for the 0/1 knapsack problem, Methodologies for Intelligent Systems, (ISMIS’94), Lecture Notes in Computer Science, Vol. 869, pp. 134–143, 1994.

- [Michalewicz1996] Z. Michalewicz, "Genetic algorithms + data structures = evolutionary programs". Third, revised and extended edition, Spring, 1996.
- [MinisterioAgricultura2008a] Ministério da Agricultura, Pecuária e Abastecimento, "Produção mensal de cana, açúcar e álcool - Brasil". Disponível em: <http://www.agricultura.gov.br/pls/portal/url/ITEM/1E601535D7B1DDE9E040A8C075024A04>. Acessado em maio de 2008.
- [MinistérioAgricultura2008b] Ministério da Agricultura, Pecuária e Abastecimento, "Agronegócio Brasileiro: Uma Oportunidade de Investimentos". Disponível em <http://www.agricultura.gov.br/>. Acesso em junho de 2008.
- [Minsky1969] M. Minsky, S. Papert, "Perceptrons: an introduction to computational geometry". MA: MIT Press, Cambridge, 1969.
- [Morin1976] T. L. Morin, R. E. Marsten, "An algorithm for nonlinear knapsack problems", Management Science, 22, pp. 1147-1158, 1976.
- [Mumford2003] C. L. Mumford (Valenzuela), "Comparing Representations and Recombination Operators for the Multi-Objective 0/1 Knapsack Problem", in Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003), Volume 2, pp. 854--861, IEEE Press, Canberra, Australia, December 2003.
- [O'Brien2001] J. A. O'Brien "Sistemas de informação e as decisões gerenciais na era da Internet". Título original: Introduction to information system. São Paulo: Saraiva, 2001.
- [Obayashi1998] S. Obayashi, S. Takahashi, Y. Takeguchi, "Niching and elitist models for mogas". In A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel (Eds.), Fifth International Conference on Parallel Problem Solving from Nature (PPSN-V), Berlin, Germany, pp. 260–269. Springer, 1998.
- [O'Connor2007] J. J. O'Connor, E. F. Robertson, "Hausdorff biography". Disponível em: <http://www-history.mcs.st-andrews.ac.uk/Biographies/Hausdorff.html>. Acesso em: 14 jul. 2007.
- [Oliveira2006] F. R. S. Oliveira, D. F. Pacheco, A. Leonel and F. B. Lima Neto, "Intelligent support decision in sugarcane harvest", Proceedings of IV World Congress on Computers in Agriculture WCCA2006, pp. 456-462, Orlando - USA, 2006.
- [Oliveira2007] F. R. S. Oliveira, D. F. Pacheco, F. B. L. Neto, "Hybrid intelligent suite for decision support in sugarcane harvest", Anais do Sexto Congresso Brasileiro de Agro-Informática (SBIAgro) São Pedro, Brasil, 2007.
- [Oliver1987] I. M. Oliver, D. J. Smith and J.R.C. Holland, "A study of permutation crossover operators on the traveling salesman problem", Genetic Algorithms and their Applications: Proceedings of the Second International Conference on Genetic Algorithms, pp. 224–230, 1987.
- [Olsen1993] A. L. Olsen, "Genetic algorithms and the knapsack problem: a study of penalty functions". Masters thesis, University of North Carolina at Charlotte, 1993.

- [Ozyczka1995] A. Ozyczka, S. Kundu, "A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm", *Structural Optimization* 10 (2), pp. 94-99, 1995.
- [Pacheco2005] D. F. Pacheco, F. S. Regueira, F. B. L. Neto, "Utilização de Redes Neurais Artificiais em colheitas de cana-de-açúcar para predição de PCC, TCH e fibra", *Revista Alcoolbrás*, S. Paulo - Brasil, v. 90, 2005, pp. 60-63.
- [Pacheco2006] D. F. Pacheco, "Suporte à decisão em sistemas inteligentes de colheitas agrícolas", Monografia de Graduação apresentada como Trabalho de Conclusão de Curso, Departamento de Sistemas e Computação, Universidade de Pernambuco, Recife, Brasil, 2006.
- [Pacheco2007] D. F. Pacheco, T. D. P. Lucas, F. B. L. Neto, "How to obtain fair managerial decisions in sugarcane harvest using NSGA-II", *Proceedings of 7th International Conference on Hybrid Intelligent Systems (HIS'07)*, Kaiserslautern, Germany, 2007.
- [Pacheco2008a] D. F. Pacheco, F. R. S. Oliveira, F. B. L. Neto, "Including multi-objective abilities in the hybrid intelligent suite for decision support". In: *International Joint Conference on Neural Networks-IJCNN (World Congress on Computational Intelligence-WCCI2008)*, Hong Kong, China, 2008.
- [Pacheco2008b] D. F. Pacheco, F. B. de Lima Neto, "A fair comparison of representations, operators and algorithms for the sugarcane harvest problem". In: *IEEE International Conference on Systems, Man, and Cybernetics - SMC2008*, Singapore, Outubro, 2008.
- [Pacheco2008c] D. F. Pacheco, T. D. P. Lucas, F. B. L. Neto, "How Preferences Affect Productivity in the Sugarcane Harvest Problem - a Comparative Study of a Two-Steps MOEA", *Proceedings of 8th International Conference on Hybrid Intelligent Systems (HIS'08)*, Barcelona, Spain, 2008.
- [Papadimitriou1982] C. H. Papadimitriou, K. Steiglitz, "Combinatorial optimization: algorithms and complexity". Prentice-Hall, Inc., 1982.
- [Pareto1986] V. Pareto, "Cours D'Economie Politique", Volume 1. Lausanne: F. Rouge, 1986.
- [Park2004] S. J. Park, C. S. Hwang and P. L. G. Vlek. "Comparison of adaptive techniques to predict crop yield response under varying soil and land management conditions". *Agricultural Systems*, 2004.
- [Parks1998] G. T. Parks, I. Miller, "Selective breeding in a multiobjective genetic algorithm". In A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel (Eds.), *Fifth International Conference on Parallel Problem Solving from Nature (PPSN-V)*, Berlin, Germany, pp. 250-259. Springer, 1998.
- [Picoli2006] M. C. A. Picoli, "Estimativa da produtividade agrícola da cana-de-açúcar utilizando agregados de redes neurais artificiais: estudo de caso usina Catanduva", *Dissertação de Mestrado*, INPE, São José dos Campos, 90p, 2006.

- [Pisinger1995] D. Pisinger, “Algorithms for knapsack problems”, Thesis (Phd) - Department of Diku, University Of Copenhagen, Copenhagen, 1995.
- [Pontes2005] P. P. B. Pontes, J. V. Rocha, R. A. C. Lamparelli, “Análise temporal de índices de vegetação como subsídio à previsão de safra de cana-de-açúcar”. In: Anais do Simpósio Brasileiro de Sensoriamento Remoto, 12, Goiania: INPE, pp. 217-224, 2005.
- [Prechelt1994] L. Prechelt, “Proben1 – a set of neural networks benchmark problems and benchmarking rules”. Technical Report 21/94, Karlsruhe University, Germany, 1994.
- [Principe2000] J. Principe, N. Euliano, C. Lefebvre, “Innovating adaptive and neural systems instruction with interactive electronic books”, Proceedings of the IEEE, special issue on engineering education, 2000.
- [Proxima2008] Próxima Datasul, PIMS. Disponível em <http://www.proxima.agr.br/portal/pims.jsp>. Acesso em junho de 2008.
- [Rao1984] S. S. Rao, “Optimization: theory and applications”, Wiley, New York, 1984.
- [Reklaitis1983] G. V. Reklaitis, A. Ravindran, K. M. Rasgdell, “Engineering optimization methods and applications”, Wiley, New York, 1983.
- [Reynolds2008] R. G. Reynolds, M. Ali, T. Jayyousi, "Mining the social fabric of archaic urban centers with cultural algorithms", *IEEE Computer Journal*, vol.41, no.1, pp.64-72, Jan. 2008.
- [Rezende2003] C. G. Rezende, “Conceitos e perspectivas em sistemas de informação e de apoio a tomada de decisão”, Portal do SERPRO, Tematec, ano IX – nº68, 2003. Disponível em <http://www.serpro.gov.br/publicacao/tematec/publicacao/tematec/2003/ttec68>. Acesso em Dezembro de 2005.
- [Rosenblatt1961] F. Rosenblatt, “Principles of neurodynamics”. Spartan Books, Washington-DC, USA, 1961.
- [Rudolph1998] G. Rudolph, “On a multi-objective evolutionary algorithm and its convergence to the pareto set”. In *IEEE International Conference on Evolutionary Computation (ICEC'98)*, Piscataway, NJ, pp. 511–516, 1998.
- [Rudorff1985] B. F. T. Rudorff, “Dados Landsat na estimativa da produtividade agrícola da cana-de-açúcar”. (INPE-3744-TDL/202). Dissertação (Mestrado em Sensoriamento Remoto) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 114p. 1985.
- [Rudorff1995] B. F. T. Rudorff, Y. E. Shimabukuro, G. T. Batista, D. Lee, "The contribution of qualitative variables to a sugarcane yield model based on spectral vegetation index". In: *Simposio Latinoamericano de Percepción Remota*, Puerto Vallarta, SELPER e SIE, v.1. pp.705 – 708, 1995.
- [Rumelhart1986] D. E. Rumelhart, G. E. Hinton, R. J. Williams, “Learning internal representations by error propagation”. In: D. E. Rumelhart and J. L. Mc-Clelland (Eds.): *Parallel Distributed Processing Vol 1*. MIT Press, Cambridge, pp. 318-362, 1986.

- [Schaffer1984] J. D. Schaffer, “Some experiments in machine learning using vector evaluated genetic algorithms”, Ph. D. Thesis, TN, Vanderbilt University, Nashville, 1984.
- [Schaffer1985] J. D. Schaffer, “Multiple objective optimization with vector evaluated genetic algorithms”. In J. J. Grefenstette (Ed.), Proceedings of an International Conference on Genetic Algorithms and Their Applications, Pittsburgh, PA, pp. 93–100, 1995. Sponsored by Texas Instruments and U.S. Navy Center for Applied Research in Artificial Intelligence (NCARAI).
- [Schott1995] J. R. Schott, “Fault tolerant design using single and multi-criteria genetic algorithms”, Master’s Thesis, Boston, MA: Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.
- [Schrijver1986] A. Schrijver, “Theory of linear and integer programming”, Wiley-Interscience, New York, NY, 1986.
- [Schuler2007] W. H. Schuler; C. J. A. Bastos Filho, R. S. Alcantara, A. L. I. de Oliveira, "A hybrid hopfield network-simulated annealing approach to optimize routing processes in telecommunications networks". In Proceedings of International Conference on Intelligent Systems Design and Applications (ISDA 2007), Rio de Janeiro, 2007.
- [Simon1960] H. A. Simon, “The new science of management decisions”, Editora Harper & ROW, Nova Iorque, 1960.
- [Sinha1998] P. Sinha. “A symmetry perceiving adaptive neural network and facial image recognition”, Forensic Science International, Vol. 98, Issues 1-2, pp. 67-89, 1998.
- [Sprague1991] R. H. Sprague, H. J. Watson, “Sistemas de apoio à decisão: colocando a teoria em prática”. Editora Campus, Rio de Janeiro, 1991.
- [Srinivas1994] N. Srinivas, K. Deb, “Multi-objective function optimization using non-dominated sorting genetic algorithms”, Evolutionary Computation Journal 2 (3), pp.221-248, 1994.
- [Stair1999] R. M. Stair, G. W. Reynolds, “Principles of information systems: a managerial approach”. 4th. Course Technology Press, 1999.
- [Stair2005] R. M. Stair, G. W. Reynolds, "Fundamentals of information systems", 4th. Edition, Prentice Hall, 2005.
- [Steuer1986] R. E. Steuer, “Multiple criteria optimization: theory, computation, and application”, Wiley, New York, 1986.
- [Stuttgart2008] University of Stuttgart, “Stuttgart Neural Network Simulator” Disponível em: <http://www.nada.kth.se/~orre/snns-manual/>. Acesso em Abril 2008.
- [Tamaki1996] H. Tamaki, H. Kita, S. Kobayashi, “Multi-objective optimization by genetic algorithms: a review”. In Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC’96), Piscataway, NJ, pp. 517–522, 1996.

- [Trigo2005] T. R. Trigo, P. C. Batista Jr., F. B. de Lima Neto, “Redes neurais artificiais em colheita de cana-de-açúcar”, In: V Congresso Brasileiro de Agroinformática, 2005, Londrina. Agronegócio, Tecnologia e Inovação. Universidade Estadual de Londrina, 2005.
- [Turban2006] E. Turban, J. E. Aronson, T. P. Liang and R. Sharda, "Decision support and business intelligence systems", 8th. Edition, Prentice Hall, New York, USA, 2006.
- [UNICA2006] União da Agroindústria Canavieira de São Paulo (ÚNICA), “Geração descentralizada de emprego e renda”. Disponível em: http://www.unica.com.br/pages/sociedade_mercado1.asp. Acesso em fev/2006.
- [Valenzuela1997] M. Valenzuela-Rendón, E. Uresti-Charre, “A non-generational genetic algorithm for multiobjective optimization”. In T. Bäck (Ed.), Proceedings of the Seventh International Conference on Genetic Algorithms, San Francisco, California, pp. 658–665. Morgan Kaufmann, 1997.
- [Veldhuizen1998] D. A. V. Veldhuizen, G. B. Lamont, “Evolutionary computation and convergence to a pareto front”. In J. R. Koza, W. Banzhaf, K. Chellapilla, K. Deb, M. Dorigo, D. B. Fogel, M. H. Garzon, D. E. Goldberg, H. Iba, and R. Riolo (Eds.), Genetic Programming 1998: Proceedings of the Third Annual Conference, San Francisco, CA, pp. 22–25. Morgan Kaufmann, 1998.
- [Veldhuizen1999] D. V. Veldhuizen, “Multiobjective evolutionary algorithms: classifications, analyses, and new innovations”, PhD Thesis, Dayton, OH: Air Force Institute of Technology. Technical Report No. AFIT/DS/ENG/99-01. 1999.
- [Watannabe1992] K. Watannabe, Y. Ikeda, S. Matsuo, T. Tsuji, "Improvement of the genetic algorithm and its application". Memoirs - faculty of engineering, Fuki University, Vol. 40 (1), Japan, 1992.
- [Weber2005] L. S. Weber, "Integração de dados espectrais e indicadores meteorológicos por meio de redes neurais para a estimativa de produtividade de cana-de-açúcar". Tese (Doutorado em Engenharia Agrícola) – Faculdade de engenharia Agrícola, Universidade Estadual de Campinas, Campinas, 106p. 2005.
- [Wei2008] L. Wei, H. Zhu-hua, "Short-term power load forecasting using improved ant colony clustering". In International Workshop on Knowledge Discovery and Data Mining (WKDD 2008), pp.221-224, 2008.
- [Werbos1974] P. J. Werbos, “Beyond regression: new tools for predicting and analysis in the behavioral sciences”. Doctoral Dissertation, Applied Mathematics, Harvard University, USA, 1974.
- [Zadeh1963] L. Zadeh, "Optimality and non-scalar-valued performance criteria", Automatic Control: IEEE Transactions, Berkeley, Ca, Usa, v. 8, n. 1, pp.59-60, 1963.
- [ZanetiJr2003] L. A. Zaneti Júnior, “Sistemas de informação baseados na tecnologia web: um estudo sobre seu desenvolvimento”. Dissertação (Mestrado) – Faculdade de Economia, Administração e Contabilidade, Departamento de Administração, Universidade de São Paulo, São Paulo, 2003.

- [Zhou2006] Aimin Zhou, Yaochu Jin, Qingfu Zhang, B. Sendhoff, E. Tsang, "Combining Model-based and Genetics-based Offspring Generation for Multi-objective Optimization Using a Convergence Criterion", *Evolutionary Computation*, 2006. CEC 2006. IEEE Congress Transactions on, pp. 892-899, 16-21 July 2006.
- [Zitzler1998] E. Zitzler and L. Thiele, "Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Case Study". *Parallel Problem Solving from Nature - PPSN-V*, pp. 292-301, Sept. 1998.
- [Zitzler1999a] E. Zitzler, "Evolutionary algorithms for multiobjective optimization: methods and applications". PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 1999.
- [Zitzler1999b] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach", *IEEE Transactions on Evolutionary Computation*, 3(4), pp. 257–271, 1999.
- [Zitzler2000] E. Zitzler, K. Deb, L. Thiele, "Comparison of multiobjective evolutionary algorithms: empirical results", *Evolutionary Computation Journal* 8 (2), pp. 125-148, 2000.
- [Zitzler2001a] E. Zitzler, K. Deb, L. Thiele, C. A. C. Coello, D. Corne (Eds), "Evolutionary multi-criterion optimization", *Lecture Notes in Computer Science*, Springer, Heidelberg, 1993.
- [Zitzler2001b] E. Zitzler, M. Laumanns, L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm", Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.

Apêndice A – Trabalhos relacionados com o tema desta dissertação publicados pelo autor

I.

Título

A fair comparison of representations, operators and algorithms for the sugarcane harvest problem.

Autores: **Diogo F. Pacheco** and Fernando B. de Lima Neto

Evento: Proceeding of IEEE International Conference on Systems, Man, and Cybernetics - SMC2008, Singapore, Outubro, 2008

Abstract: This paper instantiate the sugarcane harvest problem as a multiple knapsack problem incorporating logistic data in its formulation. Different combinations of data representations, genetic operators and multi-objective (MO) evolutionary algorithms to solve the problem are evaluated. The proposed approach produced results that considered aspects such as output quality (*i.e.* relevance to decision maker), solutions spread and run-time. Tests carried out have used real data from two sugarcane mills. Finally, a MO interpretation of generated results is also suggested.

II.

Título

How Preferences Affect Productivity in the Sugarcane Harvest Problem – a Comparative Study of a Two-Steps MOEA

Autores: **Diogo F. Pacheco**, Tarcísio Daniel P. Lucas and Fernando B. Lima Neto

Evento: Proceedings of 8th International Conference on Hybrid Intelligent Systems (HIS'08), Barcelona, Spain, 2008.

Abstract: In this paper we propose a two-level MOEA to help on the sugarcane harvest decision support. This problem is multi-objective in nature, as it contains agronomical and logistic objectives considered simultaneously. Two different sets of heuristics were used during harvest decisions, namely crisp and fuzzy prioritization schema. They are both tested and compared here with regards to effective help to decision makers – via traditional metrics and attainment to decision scenarios. Simulations show that the productivity was increased in all hypothetical scenarios investigated because of the two-level MOEA.

III.

Título

Including multi-objective abilities in the hybrid intelligent suite for decision support.

Autores: **Diogo F. Pacheco**, Flávio R. S. Oliveira and Fernando B. Lima Neto

Evento: Proceedings of International Joint Conference on Neural Networks-IJCNN (World Congress on Computational Intelligence-WCCI2008), Hong Kong, China, 2008.

Abstract: Hybrid intelligent systems (HIS) are very successful in tackling problems comprising of more than one distinct computational subtask. For instance, decision-making problems are good candidates for HIS because of their frequent dual nature. This is because supporting decision-making most often involves two phases: (i) forecasting decision scenarios and (ii) searching in those scenarios. In addition to reducing the inherent uncertainty and effort in decision making, previous works in the area of decision support have shown that some of the inconveniences of the 'Inverse Problem' can be overcome by the use of *Hybrid Intelligent Decision Suites* (HIDS). This paper extends HIDS by including a third module that deals with multi-objective (MO) tasks through Evolutionary Multi-Objective Optimization (EMOO). This EMOO module helps by creating the Pareto front for each forecast scenario produced by Artificial Neural Networks (ANN), acting here as the predictive engine of the decision support system. In order to interface better with decision makers, we use a fuzzy-heuristic module of the original HIDS. To test this concept we have applied our new approach to two distinct problems: (1) diagnosis of heart diseases (of the *proben-1* data-set) and (2) automobile feature selection (of *UCI* data-set). Results have indicated that this new ensemble of intelligent techniques enhances the quality of decision making.

IV.

Título

Including multi-objective abilities in the hybrid intelligent suite for decision support.

Autores: **Diogo F. Pacheco**, Flávio R. S. Oliveira and Fernando B. Lima Neto

Evento: Proceedings of International Joint Conference on Neural Networks-IJCNN (World Congress on Computational Intelligence-WCCI2008), Hong Kong, China, 2008.

Abstract: Hybrid intelligent systems (HIS) are very successful in tackling problems comprising of more than one distinct computational subtask. For instance, decision-making problems are good candidates for HIS because of their frequent dual nature. This is because supporting decision-making most often involves two phases: (i) forecasting decision scenarios and (ii) searching in those scenarios. In addition to reducing the inherent uncertainty and effort in decision making, previous works in the area of decision support have shown that some of the inconveniences of the 'Inverse Problem' can be overcome by the use of *Hybrid Intelligent Decision Suites* (HIDS). This paper extends HIDS by including a third module that deals with multi-objective (MO) tasks through Evolutionary Multi-Objective Optimization (EMOO). This EMOO module helps by creating the Pareto front for each forecast scenario produced by Artificial Neural Networks (ANN), acting here as the predictive engine of the decision support system. In order to interface better with decision makers, we use a fuzzy-heuristic module of the original HIDS. To test this concept we have applied our new approach to two distinct problems: (1) diagnosis of heart diseases (of the *proben-1* data-set) and (2) automobile feature selection (of *UCI* data-set). Results have indicated that this new ensemble of intelligent techniques enhances the quality of decision making.

V.

Título

**How to obtain fair managerial decisions in sugarcane harvest using
NSGA-II**

Autores: **Diogo F. Pacheco**, Tarcísio Daniel P. Lucas and Fernando B. Lima Neto

Evento: Proceedings of 7th International Conference on Hybrid Intelligent Systems (HIS'07), Kaiserslautern, Germany, 2007.

Abstract: The world's demand for sugar and particularly for renewable fuels such as ethanol requires an increase in production in sugar mills. The use of artificial neural networks (ANN) posed as a predictive core associated with the algorithm NSGA-II aims at helping decision makers to optimize the multi-objective harvest problem. This paper presents two approaches and the good results achieved as compared with other classical techniques.

VI.

Título

Hybrid intelligent suite for decision support in sugarcane harvest

Autores: Flávio R. S. Oliveira, **Diogo F. Pacheco** and Fernando B. Lima Neto

Evento: Anais do Sexto Congresso Brasileiro de Agro-Informática (SBIAgro) São Pedro, Brasil, 2007.

Abstract: This paper presents a hybrid approach to Artificial Intelligence in Decision Support Systems. The focal point was the use of Fuzzy Logic and Artificial Neural Networks for the generation of predictive decision scenarios with flexible parameter boundaries. This approach was tested via a case study concerning harvest selection of plots cultivated with sugarcane. The results have shown that the combination of the two AI techniques allowed a better tackling of complex decision problems. In the case studied, it was possible to adjust the decision problem boundaries dynamically in order to optimize the harvest logistics.

VII.

Título

HIDS: Hybrid intelligent suite for decision support

Autores: Fernando B. Lima Neto, Flávio R. S. Oliveira and **Diogo F. Pacheco**

Evento: Proceedings of Seventh International Conference on Intelligent Systems Design and Applications (ISDA), Rio de Janeiro, Brazil, 2007.

Abstract: Decision making is sometimes a hard task mainly due to data and environment complexities associated with the decision process. Without using a suitable supportive tool, this process can be slow, ineffective and error-prone. Decision support systems and techniques of artificial intelligence (AI) can be quite helpful on reducing the inherent uncertainty and effort of decision making. This work presents a suite of hybrid intelligent techniques helpful in decision making, the Hybrid Intelligent Decision Suite (HIDS). The system is composed of two complementary modules, one for forecasting new decision variables and the other, for searching among generated results of future scenarios (*i.e.* candidate decisions). Using this synergistic approach, HIDS is also suitable to obtain conditioning factors leading to desired decision, thus, overcoming some of the challenges posed by the 'Inverse Problem'. To test this concept we have applied our approach on two distinct problems: (1) diagnosis of cardiologic diseases (of the proben-1 data-set) and (2) automobile feature selection (of UCI data-set). In the simulations carried out here, the HIDS suite of Artificial Neural Networks (ANNs) and Fuzzy Logic Controllers (FLCs) were coupled to explore decision space bypassing the 'Inverse Problem' in both examples. Results proved that the ideas presented here can be effective to assemble tools which reduce uncertainty and improve quality in decision making about future scenarios.

VIII.

Título

Intelligent support decision in sugarcane harvest

Autores: Flávio R. S. Oliveira, **Diogo F. Pacheco**, Amanda Leonel and Fernando B. Lima Neto

Evento: Proceedings of IV World Congress on Computers in Agriculture WCCA2006, Orlando - USA, 2006.

Abstract: This paper presents a computing approach to support harvest decisions in sugarcane utilizing artificial intelligence (AI). The proposed two-step Decision Support System (DSS) starts with an AI indicators. Next, an heuristic that uses Genetic Algorithms (GA) is applied to search, and then recommend, suitable areas to be harvested. This work includes some experiments with real data where the ideas put forward are tested. The results of these experiments proved our approach: (i) useful to decision makers and (ii) easily coupled to current Management Information Systems (MIS) existing in most sugarcane mills.

IX.

Título

Suporte à decisão em sistemas inteligentes de colheitas agrícolas

Autores: **Diogo F. Pacheco**

Evento: Monografia de Graduação apresentada como Trabalho de Conclusão de Curso, Departamento de Sistemas e Computação, Universidade de Pernambuco, Recife, Brasil, 2006.

Resumo: Aparentemente de fácil execução, a colheita da cana-de-açúcar inclui um processo decisório muito complexo que pode refletir diretamente na produtividade e conseqüentemente, na lucratividade de toda a safra agrícola. Paradoxalmente, o grande volume informações disponibilizadas pelos sistemas de informação nas usinas sucro-alcooleiras agrega pouco valor à decisão de gestores sobre que lotes devem ser colhidos.

Utilizando-se técnicas de redes neurais artificiais para a modelagem do problema da colheita da cana-de-açúcar, este trabalho apresenta um aplicativo computacional inteligente de apoio à decisão que disponibiliza informações gerenciais, com mais precisão e antecedência (*e.g.* PCC, TCH e Fibra). Essas informações passam a alimentar um sistema de suporte à decisão (SSD) que é capaz de resolver problemas semi-estruturados de maneira interativa e rápida, através de modelos matemáticos e cenários econômicos parametrizados por seus usuários. O sistema sugere decisões de colheita.

A implementação deste sistema seguiu a metodologia proposta por [Buarque98] e com a modelagem aprimorada por [Pacheco05] realizando um desenvolvimento de uma ferramenta pioneira na área.

Espera-se que a adoção deste sistema por usinas de açúcar possa permitir ao agente de decisão conhecer o momento mais apropriado para a colheita de qualquer variedade cultivada em sua unidade.

Os resultados experimentais obtidos podem possibilitar a construção de uma ferramenta que implemente e estenda o modelo utilizado neste trabalho no trato de outras atividades relacionadas com o manejo da cana-de-açúcar ou a exportação do modelo para outras culturas agrícolas.

X.

Título

Utilização de Redes Neurais Artificiais em colheitas de cana-de-açúcar para predição de PCC, TCH e fibra

Autores: **Diogo F. Pacheco**, Filipe da S. Regueira e Fernando B. Lima Neto

Evento: Revista Alcoolbrás, S. Paulo - Brasil, v. 90, pp. 60-63, 2005.

Resumo: Aparentemente de fácil execução, a colheita da cana-de-açúcar, inclui um processo decisório muito complexo que reflete diretamente na produtividade e conseqüente, lucratividade de toda a safra agrícola. O grande volume informações disponibilizadas pelas usinas sucro-alcooleira, agregam pouco valor à decisão de gestores sobre que lotes devem ser colhidos. Utilizando-se técnicas de redes neurais para a modelagem do problema da colheita da cana-de-açúcar, disponibiliza-se antecipadamente informações gerenciais, com mais precisão e pertinência, tais como PCC, TCH e fibra. Os resultados experimentais obtidos possibilitam a construção de uma ferramenta que implemente e estenda o modelo utilizado neste trabalho para gerar curvas de maturação das variedades de cana-de-açúcar desejadas em vários cenários. Isto permitiria o agente de decisão conhecer o momento exato para a colheita de qualquer variedade cultivada em sua unidade.