



UNIPAC

UNIVERSIDADE PRESIDENTE ANTÔNIO CARLOS  
FACULDADE DE CIÊNCIA DA COMPUTAÇÃO E  
COMUNICAÇÃO SOCIAL

**CURSO DE CIÊNCIA COMPUTAÇÃO**

Emanuella da Silva Maciel

**APLICAÇÕES DE ALGORITMOS GENÉTICOS PARA OTIMIZAÇÃO DA  
LOGÍSTICA DOS CORREIOS NA ZONA DA MATA MINEIRA, UTILIZANDO O  
PROBLEMA DO CAIXEIRO VIAJANTE**

BARBACENA  
DEZEMBRO DE 2004

Emanuella da Silva Maciel

APLICAÇÕES DE ALGORITMOS GENÉTICOS PARA OTIMIZAÇÃO DA  
LOGÍSTICA DOS CORREIOS NA ZONA DA MATA MINEIRA, UTILIZANDO O  
PROBLEMA DO CAIXEIRO VIAJANTE

Monografia apresentada à Universidade  
Presidente Antônio Carlos – UNIPAC-  
Barbacena, como requisito para obtenção  
do título de bacharel em Ciência da  
Computação.

ORIENTADORA: Prof. M. Lorena Sophia Campos de Oliveira

BARBACENA  
DEZEMBRO DE 2004

Emanuella da Silva Maciel

APLICAÇÕES DE ALGORITMOS GENÉTICOS PARA OTIMIZAÇÃO DA  
LOGÍSTICA DOS CORREIOS NA ZONA DA MATA MINEIRA, UTILIZANDO O  
PROBLEMA DO CAIXEIRO VIAJANTE

Monografia apresentada à Universidade  
Presidente Antônio Carlos – UNIPAC-  
Barbacena, como requisito para obtenção  
do título de bacharel em Ciência da  
Computação.

Aprovada em \_\_\_\_/\_\_\_\_/\_\_\_\_

BANCA EXAMINADORA

Prof. M. Lorena Sophia Campos de Oliveira (Orientadora)

Universidade Presidente Antônio Carlos

Prof. José da Silva Filho

Universidade Presidente Antônio Carlos

Prof. Eduardo Macedo Bhering

Universidade Presidente Antônio Carlos

Dedico este trabalho ao meu pai,  
minha mãe, irmãs, sobrinha, vovó, tias,  
Léo, Aysha, amigos, professores e a todos  
que me ajudaram a vencer essa batalha  
quase perdida.

Agradeço a Deus, às forças nas quais acredito e ao apoio incondicional de minha família, namorado, amigos e professores, em especial a orientadora Lorena, aos professores Bhering e Silva Filho e ao coordenador do curso de Ciência da Computação Emerson Tavares.

## **RESUMO**

O trabalho apresenta uma simulação das Aplicações de Algoritmos Genéticos para Otimização da Logística dos Correios na Zona da Mata Mineira, utilizando o problema do Caixeiro viajante, baseando-se na lógica de funcionamento do Sistema GIDEON, apresentada em Arakaki (1998).

O GIDEON é um sistema de algoritmo genético que resolve o problema de roteamento de veículos e será utilizado como melhor solução para a simulação desta aplicação.

Palavras Chave: Algoritmos Genéticos, Problema do Caixeiro Viajante, Logística dos Correios, Otimização, Sistema GIDEON.

## **ABSTRACT**

The work presents a simulation of the Applications of Genetic Algorithms for Optimization of the Logistics of the Mail of the Zona da Mata Mineira using the Travelling Salesman Problem basing on the logic of operation of the Sistema GIDEON presented in Arakaki (1998).

GIDEON is a system of genetic algorithm that it solves the Problem of Itinerary of Vehicles and it will be used as better solution for the simulation of this application.

**Key Words:** Genetic Algorithms, Travelling Salesman Problem, Logistics of the Mail, Optimization, GIDEON System.

## LISTA DE FIGURAS

Figura 1: História das Abordagens de Cima para Baixo e de Baixo para Cima .....	23
Figura 2: Componentes de um Sistema de Inteligência Artificial .....	24
Figura 3: Esquema da Taxonomia dos Sistemas Naturais .....	27
Figura 4: Pseudocódigo de um AG .....	35
Figura 5: Estrutura básica de um AG .....	36
Figura 6: Método de Seleção por Roleta .....	38
Figura 7: Algoritmo básico do Método de Seleção por Roleta .....	39
Figura 8: Algoritmo básico para o Método de Seleção por Torneio .....	39
Figura 9: Cruzamento em Um Ponto .....	40
Figura 10: Cruzamento em Dois Pontos .....	41
Figura 11: Mutação Simples .....	41
Figura 12: Estrutura básica do Algoritmo de Programação Genética .....	44
Figura 13: Configuração do Encaminhamento Postal da Região da Zona da Mata Mineira ...	49
Figura 14: Divisão de Clientes utilizando Ângulos Sementes .....	50
Figura 15: Solução Viável .....	53
Figura 16: Melhor Solução .....	53
Figura 17: Divisão da Configuração Postal da Região do Campo das Vertentes utilizando Ângulos Sementes .....	56

## LISTA DE SIGLAS

AE - Algoritmos Evolucionários

AG – Algoritmo Genético

AG's – Algoritmos Genéticos

CAP – Qualificação do Curso de Computação Aplicada

CE - Computação Evolucionária

CTCE – Central de Tratamento de Coleta e Entrega

DEE – Departamento de Engenharia Elétrica

DSC – Departamento de Sistema e Computação

EE – Estratégia de Evolução

FAPEMIG – Fundação de Amparo à Pesquisa do Estado de Minas Gerais

FE – Faculdade de Engenharia

GIDEON – Sistema de algoritmo genético que resolve o problema de roteamento.

GSI – Grupo de Sistemas Inteligentes

IA – Inteligência Artificial

IC – Inteligência Computacional

ICA – Núcleo de Pesquisa em Inteligência Computacional Aplicada

INPE – Instituto Nacional de Pesquisas Espaciais (INPE)

LA – Linhas Auxiliares

LCE – Linha Coleta e Entrega

LTR – Linha Tronco Regional

MIT – Instituto de Tecnologia de Massachussets

OP – Operadores Genéticos

PCV – Problema do Caixeiro Viajante

PE - Programação Evolucionária

PG - Programação Genética

PRV – Problema de Roteamento de Veículos

PUC - Rio – Pontifícia Universidade Católica do Rio de Janeiro

SBC – Sociedade Brasileira de Computação

SE – Sistemas Especialistas

TC – Técnica Computacional

TSP – Travelling Salesman Problem

UC – Universo Científico

UERJ – Universidade Estadual do Rio de Janeiro

VA – Vida Artificial

## SUMÁRIO

1	INTRODUÇÃO .....	12
2	HISTÓRIA DA INTELIGÊNCIA ARTIFICIAL .....	17
	2.1 - Conceitos da Inteligência Artificial .....	21
	2.2 - Desenvolvimento da IA e de um Sistema de IA .....	23
3	INTELIGÊNCIA COMPUTACIONAL .....	25
	3.1 - Computação Evolutiva .....	25
	3.1.1 - Idéias Básicas da Computação Evolutiva .....	28
	3.2 - Algoritmos Evolutivos .....	29
4	ALGORITMOS GENÉTICOS .....	31
	4.1 - Histórico dos Algoritmos Genéticos .....	31
	4.1.1 - Bases Biológicas e Evolucionistas .....	31
	4.1.2 - Precursores dos AG's .....	32
	4.2 - Definições e Conceitos Básicos .....	33
	4.2.1 – População .....	37
	4.2.2 - avaliação de Aptidão .....	37
	4.2.3 – Seleção .....	38
	4.3 - Estrutura e Componentes Básicos .....	40
	4.3.1 - Cruzamento (Crossover) .....	40
	4.3.2 – Mutação .....	41
	4.4 - Parâmetros Genéticos .....	42
	4.5 – Geração .....	43
5	IMPLANTAÇÃO PROGRAMAÇÃO E USABILIDADE DOS AG'S .....	44
	5.1 - Métodos e Critérios para a Implantação de um AG .....	44
	5.2 - Operadores Genéticos .....	46
	5.3 - Critério de Término .....	46
	5.4 – Limitações .....	46
6	PROBLEMA DO CAIXEIRO VIAJANTE .....	47
7	LOGÍSTICA DOS CORREIOS .....	49
	7.1 - Configuração do Encaminhamento Postal da Região da Zona da Mata Mineira .....	49
8	SISTEMA GIDEON .....	50

<b>9</b>	SIMULAÇÃO DAS APLICAÇÕES DE ALGORITMOS GENÉTICOS PARA OTIMIZAÇÃO DA LOGÍSTICA DOS CORREIOS NA ZONA DA MATA MINEIRA, UTILIZANDO O PROBLEMA DO CAIXEIRO VIAJANTE E A LÓGICA DE FUNCIONAMENTO DO SISTEMA GIDEON .....	52
	<b>9.1</b> - Descrição do Problema .....	52
	<b>9.2</b> - AG aplicados ao Problema do caixeiro Viajante na Logística dos correios .....	56
<b>10</b>	CONCLUSÃO .....	59
<b>11</b>	REFERÊNCIAS BIBLIOGRÁFICAS .....	60
	GLOSSÁRIO .....	63
	ANEXO A - Terminologia Biológica .....	66
	ANEXO B- Inteligência Artificial ou Inteligência Computacional .....	68
	ANEXO C - O que a Inteligência Artificial pode ou não pode Fazer .....	70
	APÊNDICE A - Ensino de Computação que Ofusca as Ciências Básicas .....	73
	APÊNDICE B - Quais disciplinas devem ser Incluídas no Curso de Graduação em Ciência da Computação, de maneira que proporcione ao aluno uma boa base para trabalhar nesta área, não só como profissional, mas também para seguir um curso de pós-graduação .....	75
	APÊNDICE C - Software de Planejamento da logística de entrega e de Agendamento Reduz Custos de Distribuição .....	77
	APÊNDICE D - Inteligência Artificial: Uma forma inteligente de agendar tarefas na Volvo .....	79
	APÊNDICE E - Darwinismo Eletrônico .....	81

# 1 INTRODUÇÃO

O sonho de fabricar artefatos com propriedades inteligentes acompanham o ser humano há séculos. O mito de Frankenstein é apenas uma característica da época pré-histórica da Inteligência Artificial (FALQUETO, 2004).

O final do século XIX foi marcado por mudanças radicais na ciência. Ao lado do nascimento das geometrias não euclidianas, surgia a lógica formal com *Russel*, a descoberta do neurônio onde *Golgi* inventa um método para colorir e vê-los ao microscópio, e a traçar circuitos neurais *Cajal* dedicou sua vida. Assim foram dados os primeiros passos para a compreensão da inteligência humana.

Posteriormente *Church* com seu Lambda Cálculo e *Turing* com sua máquina, estabelecem limites para aquilo que podia ser resolvido por computadores, introduzindo o conceito de computabilidade em 1936. *Warren MMcCulloch*, juntamente com o matemático *Pitts*, propõem um modelo para o neurônio e *Hebb* descobre a plasticidade das sinapses, dando uma explicação para o aprendizado do cérebro em 1943 (FALQUETO, 2004).

Na época do aparecimento dos computadores a admiração com as inúmeras possibilidades destas máquinas fizeram com que fossem conhecidas como cérebros eletrônicos, porém no início dos anos 60 passaram a ser considerados incapazes de qualquer forma de raciocínio

Desde o surgimento do primeiro computador pessoal, pesquisadores em ciência da computação tentam livrá-lo do estigma de ser uma máquina burra (MARIA, 2004).

Cientistas em diversos países do mundo lutam contra essa limitação, na tentativa de desenvolver uma Inteligência Artificial e Computacional, que simule o funcionamento do cérebro e da mente humana, sendo esta, capaz de levar o computador a um modelo de raciocínios e sentimentos.

A Inteligência Artificial (IA) surgiu na década de 50 nos Estados Unidos, quando *John McCarthy* reuniu vários pesquisadores para estudar e propor uma nova área do conhecimento baseada nos estudos da cibernética (CITI, 2004).

A IA tem como objetivo tornar uma máquina, capaz de executar tarefas que, devido a sua natureza cognitiva, somente os seres humanos eram capazes de executar.

Nesta época, inúmeros projetos de Inteligência Artificial, já consumiam talentos de cientistas, especialmente os do MIT (Instituto de Tecnologia de *Massachussets*), onde se

encontra Marvin Minsky, considerado o pai da IA e o maior defensor da idéia de tornar as máquinas pensantes.

O computador pessoal evoluiu muito e continuará se transformando paralelamente ao esforço da Inteligência Artificial em construir raciocínios e fazer uma máquina aprender tal como uma criança, que precisa passar por uma série de aprendizados até conseguir pensar sozinha (MARIA, 2004).

Com o avanço da ciência moderna, a busca da IA tem tomado dois caminhos fundamentais: a pesquisa psicológica e fisiológica da natureza do pensamento humano e o desenvolvimento tecnológico de sistemas de informática cada vez mais complexos (CITI, 2004).

Estes caminhos têm como meta principal, a construção de uma máquina inteligente e, conseqüentemente, pesquisar a natureza da inteligência. E esta é alcançada através: da investigação de como as pessoas pensam quando estão tentando tomar decisões e resolver problemas; da divisão desses processos de pensamentos em etapas básicas e da implementação um programa que solucione o problema usando essas mesmas etapas.

As pesquisas em Inteligência Artificial geraram divisões e ramificações em várias áreas, como: lógica, robótica, probabilidade, métodos intuitivos, combinações matemáticas, representação de conhecimentos, entre outras. Nela existe um imenso leque de técnicas, cada qual com sua aplicabilidade e utilidade.

As aplicações mais populares, as ferramentas mais comuns e bem sucedidas para o desenvolvimento de sistemas ou algoritmos inteligentes são: Sistemas Especialistas, Algoritmos Genéticos, Redes Neurais e Lógica Nebulosa, além de grande dose de cálculos estatísticos.

Porém a grande dificuldade existente para que sistemas inteligentes tradicionais resolvam determinadas classes de problemas, intensificaram-se as pesquisas em outro campo da IA, a Inteligência Computacional (IC) e seus derivados.

As técnicas mais conhecidas de IC são (PALAZZO, 1997): Redes Neurais, Lógica Nebulosa (*Logic Fuzzy*), Computação Evolutiva e Agentes Autônomos.

Os sistemas inteligentes desenvolvidos atualmente têm finalidades bastante distintas, como: acionamentos eletromecânicos, planejamento de trajetórias para robôs, desenvolvimento de circuitos eletrônicos, controle de operações financeiras no mercado de valores ou reconhecimento de padrões para controle em Sistemas Especialistas.

Segundo o Professor Mário Neto Borges (apud MARIA, 2004), diretor científico da FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais) e especialista em

IA, “na simulação da mente, o Sistema Especialista supera a condição humana e quanto mais especialista maior a possibilidade de superação”. Disse também que, um Sistema Especialista reúne todo o conhecimento que um profissional precisa ter em sua área de atuação, ou seja, “O sistema raciocina como se fosse a mente deste profissional. É a possibilidade de combinar conhecimento e experiência de mais de um especialista na mesma base de conhecimento”. E, cita um exemplo: “O diagnóstico médico, uma aplicação de IA das mais difundidas em todo o mundo”.

Segundo *Marvin Minsky*, engenheiro e cientista do MIT, para conseguir tais façanhas de uma máquina, basta pedir a ela que desempenhe várias tarefas. Um exemplo: solicitar ao computador que encontre a saída de um labirinto. Oferecer-lhe várias possibilidades com uma única resposta, prevendo que será necessário fazer várias tentativas até ele acertar. Mas, depois de conseguir isso, não poderá mais errar. O computador tem que ter aprendido (MARIA, 2004).

“Assim, o computador cria um sistema de memória para entrar e sair do labirinto”, afirma *Minsky* (apud MARIA, 2004).

Isto é a IA e é através dessa tecnologia que são oferecidas à máquina bilhões, de combinações para ela se concentrar em apenas uma (MARIA, 2004).

*Minsky* acredita que é possível ensinar a máquina, inclusive por meio de suas falhas e cita o caso clássico do computador da *IBM Deep Blue* que foi construído para jogar com o campeão de xadrez *Gary Kasparov*. “A máquina já perdeu do jogador e através de suas falhas foi possível melhorar as combinações do *Deep Blue*”.

E frisa que, “neste caso, ainda não existe inteligência no computador da IBM, ele foi apenas bem programado para jogar xadrez”, e almeja o computador com gerenciadores de programas que permitam decidir. Este é outro estágio da IA, que usa mais informações e tenta representar o bom senso por meio do conhecimento. “Se sabemos como o cérebro funciona, podemos simular através da IA”, lembra o cientista.

Os estudos em IA estão passando rapidamente dos laboratórios para a aplicação prática, onde estão sendo desenvolvidas máquinas capazes de realizar tarefas como planejamento estratégico ou a aprendizagem por experiência.

Mas, a dificuldade para que sistemas resolvam determinados problemas complexos como o diagnóstico de falhas em aviões, satélites ou o controle de grandes fábricas e departamentos, estão cada vez mais a cargo da Inteligência Computacional.

Técnicas de IC, inspiradas em biologia, vêm sendo usadas em diversas áreas. Aplicações genéticas foram inicialmente usadas na década de 80, concentrando-se na

otimização do posicionamento de componentes e rotas para trilhas de circuitos integrados. Porém, no início da década de 90, pesquisadores passaram a aplicá-las, no desenvolvimento de arquitetura, projeto de circuitos que evoluem por conta própria e para a resolução de problemas de alta complexidade. Estes utilizam técnicas vindas da computação evolutiva, como Algoritmos Genéticos e Programação Genética.

Os Algoritmos Genéticos (AG's) foram introduzidos por *John Holland*, na década de 70, com o intuito de aplicar as idéias de Seleção Natural e Evolução Biológica propostas por *Charles Robert Darwin*, ou seja, utilizar desses conceitos para buscar melhorias na Teoria Computacional (YEPES, 2002).

Diferentemente dos métodos de busca e otimização tradicionais, AG trabalham não com uma única solução, mas com uma população de soluções onde, apresenta um grupo de soluções candidatas (população) que, por seleção natural, operadores genéticos, mutação e cruzamento, encontram os cromossomos (indivíduos) com melhores aptidões. A seleção natural garante que os cromossomos mais aptos gerem descendentes (soluções cada vez melhores) nas populações futuras (MIRANDA, 1999).

Devido a seu funcionamento, este trabalho para conclusão de curso tem por finalidade mostrar a evolução, características e ramificações da Inteligência Artificial e Computacional, bem como o funcionamento e a programação dos Algoritmos Genéticos que, por obterem soluções aproximadamente ótimas, seguindo regras baseadas em genética, serão combinados ao Problema de Roteamento e Planejamento do Caixeiro Viajante.

O Problema do Caixeiro Viajante (PCV) pode ser definido como o problema para encontrar um roteiro de menor distância ou custo que passa por um conjunto de cidades, podendo cada cidade, ser visitada apenas uma vez (BOAVENTURA, 2001).

Os AG's e o PCV, terão ênfase, quando aplicados à Logística dos Correios, visando analisar a melhor técnica para otimização em tempo e distância na entrega de correspondências.

Para a simulação desta aplicação, avaliou-se o funcionamento do Sistema GIDEON (sistema de algoritmo genético para resolver o problema de roteamento de veículos que consiste em dois módulos, um de agrupamento global e outro de otimização local de rotas) e o encaminhamento postal da Região da Zona da Mata Mineira, onde estão compreendidas as cidades de Belo Horizonte, Barbacena, Juiz de Fora, Leopoldina, Barroso, São João del Rei, Resende Costa, Prados, Dolores de Campos, Carandaí, Conselheiro Lafaiete, Congonhas, Ouro Branco, Ewbank da Câmara, Santos Dumont, Rio Pomba, Tocantins, Ubá, Visconde do Rio

Branco, Bicas, São João Nepomuceno, Guarani, Piraúba, Astolfo Dutra, Muriaé, Carangola, Além Paraíba e Santo Antônio do Aventureiro.

O trabalho apresentará no capítulo 2 a história, conceitos e desenvolvimentos da IA e do Sistema de Inteligência Artificial; No capítulo 3 será descrito um campo da Inteligência Artificial, ou seja, a Inteligência Computacional e seus derivados; No capítulo 4 serão abordados os Algoritmos Genéticos, com suas bases biológicas, Precusores, Conceitos, Estrutura, Componentes Básicos, Parâmetros e Operadores Genéticos; O capítulo 5 mostrará a implantação, programação e usabilidade dos algoritmos genéticos; O capítulo 6 descreve o Problema do Caixeiro Viajante; O funcionamento da Logística dos Correios na Zona da Mata Mineira será mostrado no capítulo 7; O capítulo 8 mostrará o funcionamento do Sistema GIDEON. O capítulo 9 abordará a simulação das aplicações de algoritmos genéticos para otimização da logística dos correios, utilizando o problema do caixeiro viajante a partir do Sistema GIDEON; No capítulo 10 serão apresentadas as conclusões do trabalho. As Referências Bibliográficas estão contidas no capítulo 11, onde após seguem o glossário, para melhor compreensão textual, alguns anexos e apêndices que complementam e enriquecem o trabalho.

## 2 HISTÓRIA DA INTELIGÊNCIA ARTIFICIAL

Apesar de relativamente recente, esta ciência é a realização do sonho humano que remonta a Antiguidade Clássica. No Renascimento, e com a expansão de um espírito prático e quantitativo, surge a mecânica e com ela, uma nova concepção do homem (CITI, 2004).

O estudo da Inteligência Natural é bastante antigo, a mais de dois mil anos os filósofos têm tentado entender como visão, o aprendizado, a memória e o raciocínio do homem poderiam ser obtidos.

Imprescindíveis para o avanço da Inteligência Artificial (IA), foram os trabalhos dos cientistas e engenheiros dos séculos XVII a XIX, que construíram as primeiras calculadoras e robôs mecânicos (SANTOS, 2003).

Porém, nenhuma daquelas máquinas primitivas tinha um programa tão ambicioso quanto os esforços contemporâneos em relação a IA (LAUDON, 1999), surgindo então a figura do cientista britânico Alan Turing. Em 1956 John McCarthy criou oficialmente o termo Inteligência Artificial, que passou a ser reconhecida como ciência.

Os desenvolvimentos em IA caminham lado a lado com a Evolução Computacional, que ao longo do tempo tenta encarar máquinas como inteligentes.

No entanto seu objeto de estudo continua semicoberto, pois o homem ainda não possui uma definição satisfatória de inteligência e, para compreender os processos da inteligência artificial e da representação do conhecimento terão que dominar os conceitos da inteligência humana (CITI, 2004).

Nos últimos anos tem-se dado atenção a alguns setores de pesquisa abandonados no passado, como a representação de redes neurais e a lógica nebulosa, interesses renovados graças aos enormes progressos a que se tem assistido no domínio da ciência da computação.

A Inteligência Artificial é o ramo do conhecimento humano que se propõe a entender e a construir entidades inteligentes que apresentem as mesmas capacidades das entidades inteligentes encontradas no mundo real (SANTOS, 2003). “É a propriedade de um artefato de poder resolver problemas que se fossem resolvidos por um ser vivo seria considerado inteligente” (BARRETO, 1997 apud FALQUETO, 2004).

Atualmente, a IA compreende uma enorme variedade de subcampos, abrangendo desde áreas de propósito gerais como: percepção e raciocínio lógico, até tarefas específicas como: jogar xadrez, provar teoremas matemáticos, escrever poesias, e diagnosticar doenças (SANTOS, 2003).

Freqüentemente, cientistas de outros campos se movem para dentro da IA, pois encontram ferramentas e vocabulários, para sistematizar e automatizar as tarefas intelectuais nas quais eles vinham trabalhando. Semelhantemente, pesquisadores em IA podem escolher onde aplicar seus métodos e em que área de interesse intelectual humano.

De acordo com Russell e Norvig (1995 apud SANTOS, 2003), podemos classificar as abordagens ao campo de Inteligência Artificial em quatro direções:

- Sistemas Computacionais que *pensam* como os humanos: *a abordagem da modelagem cognitiva*.

**“Os recentes e excitantes esforços realizados para fazer com que os computadores pensem... máquinas com mente, no sentido pleno e literal” (Haugeland, 1985).**

**“A automação de atividades que associamos com o pensamento humano, atividades tais como tomada de decisão, resolução de problemas, aprendizagem...” (Bellman, 1978).**

“Se vamos dizer que um programa pensa como os humanos então temos que saber como os humanos pensam. Precisamos adentrar as operações reais da mente humana e existem duas maneiras de fazer isso: através da introspecção (procurando capturar nossos próprios pensamentos à medida que eles ocorrem) ou através de experimentos psicológicos. Uma vez que tenhamos uma teoria suficientemente precisa da mente, torna-se possível expressar a teoria, como um programa de computador” (SANTOS, 2003).

Newell e Simon (1961 apud SANTOS, 2003) e o programa GPS – General Problem Solver, queriam comparar os passos dados pelos humanos e pelo programa para resolver problemas, e não obter a resposta correta como queriam outros pesquisadores.

O campo interdisciplinar da ciência cognitiva junta modelos computacionais da IA e técnicas experimentais da psicologia para tentar construir teorias precisas e testáveis de como trabalha a mente humana, são eles:

- Sistemas Computacionais que *agem* como os humanos: *a abordagem do Teste de Turing*.

**“A arte de criar máquinas que executam funções que requerem inteligência quando executadas pelas pessoas” (Kurzweil, 1990).**

**“O estudo de como fazer os computadores executarem coisas as quais, no momento, as pessoas executam melhor” (Rich e Knight, 1991).**

Em 1948, Alan Turing propôs um teste, o chamado Teste de Turing, onde um humano e um computador são colocados em compartimentos separados e interligados com o propósito de se comunicar; se a pessoa não souber dizer se está falando com um computador ou com outro ser humano, então a máquina é inteligente; mas até hoje nenhuma máquina passou nesse teste (LAUDON, 1999).

Para programar um computador para passar no teste, precisaríamos dotá-lo das seguintes capacidades: Processamento de Linguagem Natural, Representação de Conhecimento, Raciocínio Automatizado e Aprendizagem de Máquina. Se tivermos em mente o Teste de Turing total, teríamos que acrescentar: Visão por Computador e Robótica (SANTOS, 2003).

- Sistemas Computacionais que *pensam* racionalmente: *a abordagem das leis do pensamento*.

**“O estudo das faculdades mentais através do uso de modelos computacionais” (Charniak e McDermott, 1985).**

**“O estudo de computações que torna possível ao computador perceber, raciocinar e agir” (Winston, 1992).**

Em 1965 já existiam programas que permitiam a descrição do problema em notação lógica e encontravam a sua solução, se ela existisse. Pesquisadores de IA pertencentes à tradição da lógica esperam basear-se em tais programas para criar sistemas inteligentes.

O primeiro obstáculo a esta abordagem é o fato de não ser muito fácil pegar conhecimento informal e colocá-lo nos termos formais requeridos pela notação lógica. O segundo, é que existe uma grande diferença entre ser capaz de resolver um problema em princípio e fazê-lo assim na prática.

- Sistemas Computacionais que *agem* racionalmente: *a abordagem do agente racional*

"Um campo de estudo que procura explicar e emular o comportamento inteligente em termos de processos computacionais" (Schalkoff, 1990).

"O ramo da ciência da computação que se ocupa com a automação do comportamento inteligente" (Luger e Stubblefield, 1993).

Agir racionalmente significa agir de maneira a obter as metas de alguém, uma vez fornecidas as crenças dessa pessoa. Um agente é exatamente alguma coisa que percebe e atua. Nesta abordagem, IA é vista como o estudo e construção de agentes racionais.

Analisando a IA através das "leis do pensamento", a ênfase era colocada nas inferências corretas. Fazer inferências corretas é algumas vezes parte do fato de ser um agente racional.

Por outro lado, inferência correta não é tudo sobre racionalidade, porque existem situações onde não há uma coisa correta provável a se fazer, e ainda assim alguma coisa precisa ser feita.

Há também formas de se agir racionalmente da qual não se pode dizer de maneira razoável que envolva inferência.

Todas as habilidades cognitivas necessárias para o Teste de Turing são necessárias para permitirem ações racionais (SANTOS, 2003):

- **Habilidade de representar o conhecimento e raciocinar com ele:** porque isso nos capacita alcançar boas decisões em uma ampla variedade de situações;
- **Habilidade de gerar sentenças compreensivas em linguagem natural:** porque dizer aquelas sentenças ajuda a nos relacionar numa sociedade complexa;
- **Habilidade de aprender não apenas por erudição:** porque tendo uma idéia melhor de como o mundo funciona nos capacita a gerar estratégias mais eficazes para lidar com ele;
- **Habilidade de percepção visual:** porque assim teremos uma melhor idéia sobre o que está envolvido numa ação.

O estudo de IA segundo a abordagem dos agentes racionais tem duas vantagens:

- Mais geral que a abordagem das leis do pensamento: porque a inferência correta é apenas um mecanismo útil para obter racionalidade, e não algo necessário;
- Mais adequada de ser testada pelo desenvolvimento científico do que abordagens baseadas no comportamento humano ou no pensamento humano: porque o padrão da racionalidade é claramente definido e completamente geral.

Deve-se ter em mente a obtenção da racionalidade perfeita, e sempre fazer a coisa certa, não é possível em ambientes complicados.

Assim a história da IA é povoada de diferentes paradigmas que se contrapõem, de teorias que defendem e abandonam, e que são consecutivamente retomadas.

## **2.1 – Conceitos de Inteligência Artificial**

Parte da ciência da computação voltada ao desenvolvimento de sistemas de computadores, aos quais exibem características associadas à inteligência humana, quanto ao seu comportamento, compreensão de linguagens, aprendizado, racionalizando e resolvendo problemas (RICH, 1994).

A Inteligência Artificial, como o próprio nome indica, permite que o computador pense, ele imita o processo básico de aprendizado humano por meio da qual novas informações são absorvidas e se tornam disponíveis para referências futuras.

A mente humana pode incorporar novos conhecimentos sem alterar seu funcionamento e sem atrapalhar todos os outros fatos que já estão armazenados no cérebro, um programa IA tem que funcionar do mesmo modo.

Existem vários modos de se compreender a IA:

- Compreender os princípios que permitem simular a inteligência humana por meio da criação de modelos computacionais de processos cognitivos;
- Desenvolver sistemas (hardware / software) mais úteis com capacidade de dedução e percepção.

A IA é hoje, um domínio do conhecimento cada vez mais “na moda”. Dela fala-se, escreve-se, ouve-se falar, lê-se. Ela não é um fenômeno isolado, mas sim uma família de atividades em que cada uma busca capturar algum aspecto da inteligência dos seres humanos e de seu modo de ver (LAUDON, 1999), tentando assim construir um campo de explicações algorítmicas para esses processos mentais.

É isto o que distingue a IA dos outros campos de saber, pois coloca ênfase na elaboração de teorias e modelos da Inteligência para programas de computador. Ela inclui ainda vários componentes chaves, como Sistemas Especialistas, Robótica, Sistemas Visuais, Processamento da Linguagem Natural, Lógica Nebulosa ou Difusa, Softwares Inteligentes e Algoritmos Genéticos, ao qual será enfatizado.

A Inteligência Artificial utiliza abordagens de otimização, convencional e heurísticas (STAIR, 2002):

- O modelo de otimização: visa encontrar a melhor solução, sendo considerado como o que melhor ajuda a alcançar metas;
- O modelo Convencional: corresponde ao que irá encontrar uma boa, mas não necessariamente a melhor solução do problema. Este modelo é geralmente usado porque a correta modelagem do problema para se obter uma decisão otimizada pode ser muito difícil, complexa ou cara. Essa abordagem normalmente não procura todas as soluções possíveis, mas somente aquelas que provavelmente darão bons resultados;
- A Heurística: é muito usada na tomada de decisão, freqüentemente está relacionada aos “princípios básicos”, diretrizes ou procedimentos comumente aceitos que, em geral, levam a uma boa solução.

Prover suporte às abordagens de otimização, convencionais e heurísticas para problemas pequenos os sistemas de suporte a decisão podem encontrar a melhor solução (otimizada); já para problemas mais complexos, são usadas as abordagens convencionais e heurísticas, ou seja, onde o computador determina uma solução muito boa, mas não necessariamente a melhor (STAIR, 2002).

## 2.2 – Desenvolvimento da IA e do Sistema de Inteligência Artificial

A IA é usada para descrever os computadores com capacidade de imitar ou duplicar as funções do cérebro humano (STAIR, 2002).

As pesquisas em IA têm sido na verdade conduzidas em duas direções, cada uma delas com sua própria história. A figura 1 mostra uma breve história dessas abordagens:

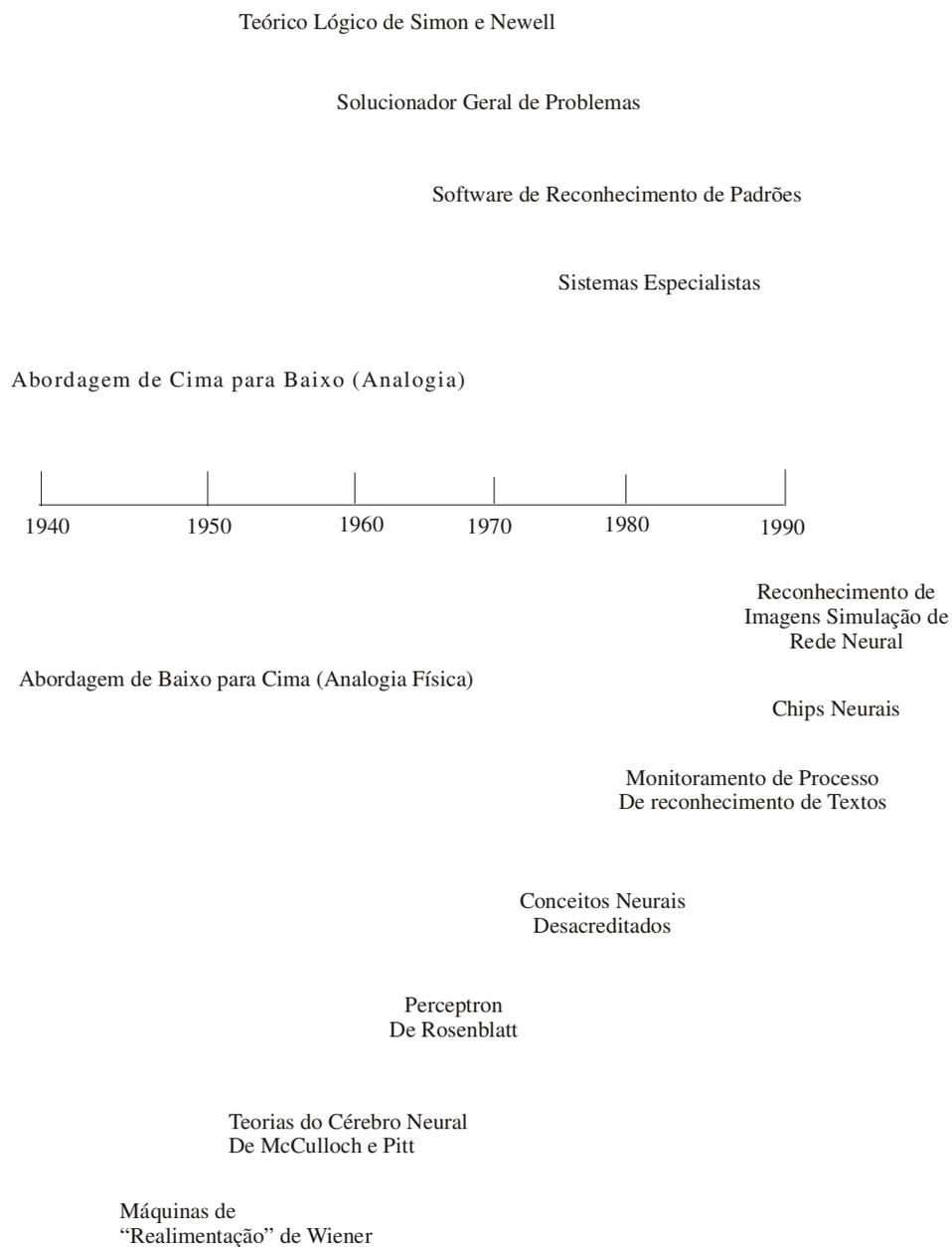


Figura 1 – História das Abordagens de Cima para Baixo e de baixo para Cima da IA

Fonte: LAUDON, 1999.

Desde a segunda Guerra Mundial, houve dois impulsos principais na pesquisa da IA. A abordagem de cima para baixo (*top-down*) busca desenvolver um modelo lógico da inteligência e do funcionamento do cérebro humano. A escola de baixo para cima (*bottom-up*) tenta construir um dispositivo físico análogo ao cérebro humano e, dessa maneira, reproduzir os padrões de pensamento humano. Ambas técnicas tiveram um papel importante nas atuais pesquisas em IA (LAUDON, 1999).

Todos os elementos nos quais consiste o processo humano de tomada de decisão, objetivos, fatos, regras, mecanismos de inferência e poda devem ser reunidos em um programa de computador para que ele possa ser realmente qualificado como um programa que possui inteligência artificial.

Daí o objetivo de desenvolver Sistemas de Inteligência Artificial, não para substituir completamente a tomada de decisão humana, mas duplicá-la para certos tipos de problemas bem definidos (STAIR, 2002).

A figura 2 demonstra os componentes de um sistema de IA baseado em regras:

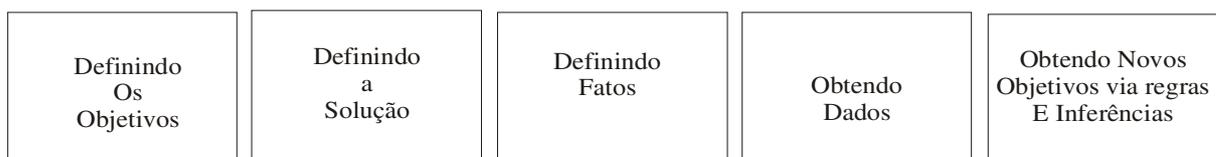


Figura 2: Componentes de um Sistema de Inteligência Artificial.

Fonte: RICH, 1994.

Essa estrutura é baseada na identificação de seus diferentes componentes, ela é o fator mais importante para tornar a programação de IA superior à programação comum.

Se, identificamos os elementos que formam os processos do pensamento humano, podemos então transferir a maneira segundo a qual pensamos a respeito de qualquer problema, para um programa de IA.

Podemos também determinar o que nossas mentes fazem em um estágio de qualquer tomada de decisão e, facilmente encontramos nesse projeto de programa, uma seção que corresponda a um aspecto equivalente da Inteligência, além de podermos fazer modificações em uma ou mais partes de um projeto, sem desarranjar a estrutura do sistema de raciocínio.

Existem também os Sistemas de Inteligência Artificial Híbridos que combinam diferentes tecnologias de IA como, algumas funções dos sistemas especialistas com funções dos AG's, AG's com redes neurais ou redes neurais com lógica difusa ou nebulosa (LAUDON, 1999).

### 3 INTELIGÊNCIA COMPUTACIONAL

A inteligência artificial, tradicionalmente, trabalha com base de dados e base de regras, realizando inferências lógicas até convergir para uma conclusão. Este processo é caracterizado como comportamento inteligente, assim como a tarefa executada por seres humanos (RICH, 1994).

Devido à dificuldade existente para que sistemas inteligentes tradicionais resolvam determinadas classes de problemas, intensificaram-se as pesquisas no subcampo da Inteligência Artificial, a Inteligência Computacional (IC).

As técnicas mais conhecidas de IC são (PALAZZO, 1997):

- Redes Neurais artificiais;
- Lógica *Fuzzy* (nebulosa);
- Computação Evolutiva;
- Agentes Autônomos.

O objeto de estudo deste trabalho, são os Algoritmos Genéticos, e se encontram inseridos no item Computação Evolutiva.

#### 3.1 – Computação Evolutiva (CE)

A Computação Evolutiva (CE) é um ramo da ciência da computação que propõe um paradigma alternativo ao processamento de dados convencional. Este, não exige para resolver um problema, o conhecimento prévio de uma maneira de encontrar uma solução.

É baseada em mecanismos evolutivos encontrados na natureza, tais como a auto-organização e o comportamento adaptativo (BITTENCOURT, 2003). Estes mecanismos foram descobertos e formalizados por *Darwin* em sua *Teoria da Seleção Natural*, segundo a qual, a vida na terra é o resultado de um processo de seleção, pelo meio ambiente, dos mais aptos e adaptados, e por isto mesmo com mais chances de se reproduzirem.

A diversidade da vida, associada ao fato de que todos os seres vivos compartilham uma bagagem genética comum, pelo menos em termos de seus componentes básicos, é um exemplo eloquente das possibilidades do mecanismo de evolução natural.

A Computação Evolutiva surgiu no final dos anos 60, quando John Holland começou a estudar a possibilidade de incorporar os mecanismos naturais de seleção e sobrevivência para a resolução de problemas de inteligência Artificial (IA), aos quais já tinham encontrado solução na natureza, mas não apresentavam uma abordagem satisfatória em sistemas computacionais (RICH, 1994).

Como resultado de sua pesquisa, Holland lançou seu livro *Adaptation in Natural and Artificial Systems* que é considerado o ponto de partida da CE.

O termo de Computação Evolutiva engloba um amplo leque de técnicas de resolução de problemas complexos de busca e aprendizagem que se baseiam na simulação dos processos descritos nas teorias da evolução, cujo objetivo fundamental não se trata de reproduzir os fenômenos que ocorrem na natureza, mas sim de tirar proveito das idéias por trás deles.

Devido ao alto esforço computacional exigido, a utilização prática de sistemas com técnicas de CE, só foram viabilizadas a partir dos anos 80 com o surgimento de computadores de alto desempenho (para a época) e baixo custo, sendo a partir de então, amplamente utilizada para resolver com êxito certos problemas de engenharia e das ciências sociais que até então não encontravam forma prática de implementação em modelos computacionais.

Deste período são os trabalhos de Goldberg e Davis (RICH, 1994), aos quais propõe soluções para problemas da vida real. Provavelmente, devido aos trabalhos, à divulgação destes autores e às novas possibilidades apresentadas, o crescimento da CE foi impressionante, tendo na atualidade inúmeras aplicações funcionando com êxito, como, por exemplo, na área industrial, onde atuam em controles de redes de distribuição de gás até o desenho de turbinas e o cálculo de estratégias de mercado (isso para citar uma pequena parte).

A CE baseia-se fundamentalmente no uso dos Algoritmos Evolutivos (AE's), cujo propósito é conduzir uma busca estocástica, fazendo evoluir um conjunto de estruturas e selecionando de modo iterativo as mais adequadas.

Um AE é considerado mais eficiente quanto melhor seja seu desempenho na solução de um determinado problema, independente de sua fidelidade aos conceitos biológicos. Na verdade, a maioria dos algoritmos que seguem este enfoque são extremamente simples do ponto de vista biológico, mas, mesmo assim, se apresentam como poderosas e eficientes ferramentas de busca.

A Computação Evolutiva divide-se em cinco paradigmas básicos: Algoritmos Genéticos, Estratégias Evolutivas, Programação Evolutiva, Sistemas Classificadores e Programação Genética, como podemos visualizar na figura 3.

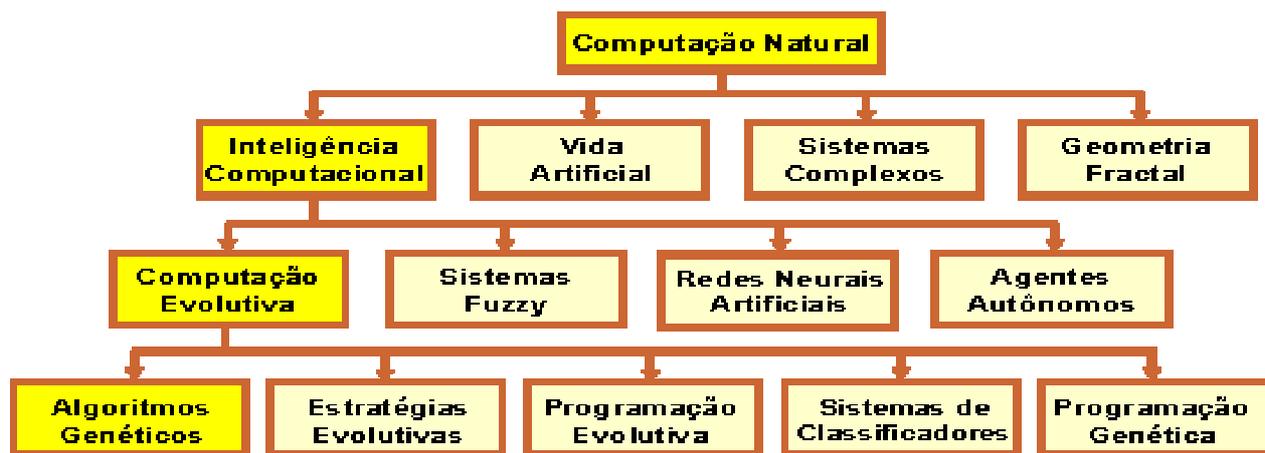


Figura 3 – Esquema da Taxonomia dos Sistemas Naturais.  
 Fonte: PALLAZZO, 1997.

Bittencourt (2003) descreve que os Algoritmos Genéticos (AG's), implementavam populações de indivíduos contendo um genótipo, formado por cromossomos (que neste modelo eram representados por cadeias de "bits"), aos quais se aplicavam operadores de seleção, recombinação e mutação. Ainda que Holland tenha proposto um quarto operador (a inversão), este não chegou a ser largamente utilizado.

Uma das primeiras aplicações propostas para os algoritmos genéticos foram os sistemas classificadores, que são sistemas de produção e, na verdade, usam os algoritmos genéticos em uma parte do algoritmo global. Apesar do interesse que levantaram na época, os sistemas classificadores permanecem como um campo de estudo em grande parte ainda inexplorado.

Outro ramo descendente dos algoritmos genéticos é o da Programação Genética (PG). Nela os indivíduos da população não são seqüências de "bits", mas sim programas de computador armazenados sob determinada forma.

Tais programas é que são os candidatos à solução do problema proposto. A Programação Genética não usa o operador mutação e a recombinação se dá pela troca de subárvores entre dois indivíduos candidatos à solução.

Outro ramo da CE é a *Programação Evolutiva* (PE), que visa prever o comportamento de máquinas de estado finitas. Apenas dois operadores são usados: a seleção e a mutação, e não foram muito consideradas na comunidade de CE por rejeitar o papel fundamental da recombinação (BITTENCOURT, 2003).

Finalmente, um último ramo é a *Estratégia Evolutiva* (EE), proposta nos anos 60, na Alemanha. A ênfase aqui é na auto-adaptação. O papel da recombinação é aceito, mas como operador secundário.

Embora tenham origens bastante diversas, todas essas abordagens têm em comum o modelo conceitual inicial (evolução natural), além dos operadores e o mesmo objetivo final: a solução de problemas complexos.

### **3.1.1- Idéias Básicas da Computação Evolutiva**

Segundo Pozo (2000) a Computação Evolutiva ou Evolucionária (CE) é um ramo de pesquisa emergente da Inteligência Artificial que propõe um novo paradigma para solução de problemas inspirado na Seleção Natural. A Computação Evolucionária compreende um conjunto de técnicas de busca e otimização inspiradas na evolução natural das espécies. Desta forma, cria-se uma população de indivíduos que vão reproduzir e competir pela sobrevivência. Os melhores sobrevivem e transferem suas características a novas gerações. As técnicas atualmente incluem Programação Evolucionária, Estratégias Evolucionárias, Algoritmos Genéticos e Programação Genética. Estes métodos estão sendo utilizados, cada vez mais, pela comunidade de inteligência artificial para obter modelos de inteligência computacional.

Algoritmos Genéticos (AG) e Programação Genética (PG) são as duas principais frentes de pesquisa em CE. Os Algoritmos Genéticos (AG) foram concebidos em 1960 por John Holland, com o objetivo inicial de estudar os fenômenos relacionados à adaptação das espécies e da seleção natural que ocorre na natureza, bem como desenvolver uma maneira de incorporar estes conceitos aos computadores.

Os AG's possuem uma larga aplicação em muitas áreas científicas, entre as quais podem ser citados problemas de otimização de soluções, aprendizado de máquina, desenvolvimento de estratégias e fórmulas matemáticas, análise de modelos econômicos, problemas de engenharia, diversas aplicações na Biologia como simulação de bactérias, sistemas imunológicos, ecossistemas, descoberta de formato e propriedades de moléculas orgânicas.

Programação Genética (PG) é uma técnica de geração automática de programas de computador, inspirada na teoria de AG's de Holland. Em PG é possível criar e manipular software geneticamente, aplicando conceitos herdados da Biologia para gerar programas de computador automaticamente.

A diferença essencial entre AG e PG é que em PG as estruturas manipuladas são bastante mais complexas, assim como várias das operações realizadas pelo algoritmo. Ambas as técnicas compartilham a mesma base teórica, inspirada na competição entre indivíduos pela sobrevivência, porém não mantêm vínculos de dependência ou subordinação. PG e AG's representam um campo novo de pesquisa dentro da Ciência da Computação. Neste campo muitos problemas continuam em aberto e a espera de novas soluções e ferramentas. Apesar disso, este paradigma vem se mostrando bastante poderoso e muitos trabalhos vêm explorando o uso de AG's e PG para solucionar diversos problemas em diferentes áreas do conhecimento desde mineração de dados e biologia molecular até o projeto de circuitos digitais e inúmeras tarefas envolvendo otimização (GECCO, 2000 apud POZO, 2000).

### **3.2 – Algoritmos Evolutivos (AE's)**

Como dito anteriormente, a Computação Evolutiva (CE) sustenta-se nos AE's, aos quais têm como finalidade a "sobrevivência do mais apto", que é conseguida por meio de "adaptação ao ambiente", ou seja, maior chance de sobrevivência dos mais aptos e estes, conseqüentemente, maior possibilidade de gerar descendentes, transmitindo as suas características genéticas.

Ao executar um AE, é criada uma população de indivíduos que nada mais são do que um conjunto de possíveis soluções para um determinado problema, os quais são submetidos a determinadas transformações (cruzamento, mutação, ...) sendo, cada ciclo completo de transformação e seleção denominado geração.

Logo, para podermos gerar um AE, devemos ter:

- Uma população de indivíduos (conjunto de soluções candidatas);
- Um processo de seleção que avalie a aptidão de cada indivíduo da população;
- Um processo de transformação, o qual gerará novas soluções a partir das existentes.

Tendo cumprido esses, pode-se realizar a implantação de um AE, cujo principal paradigma são os Algoritmos Genéticos. Essas técnicas têm sido utilizadas com resultados bastante satisfatórios em diversas técnicas como (MIT, 1996 apud RICH, 1994):

- *Otimização*: ampla variedade de tarefas de otimização, incluindo otimização numérica e problemas de otimização combinatória como o desenho de circuitos ou o problema do caixeiro viajante;
- *Programação Automática*: para fazer evoluir programas de computador para tarefas específicas e para projetar outras estruturas computacionais (como autômatos celulares);
- *Aprendizado Artificial*: para muitas aplicações de aprendizado artificial, incluindo tarefas de classificação e predição, como predição do tempo ou de estruturas protéicas;
- *Economia*: para desenvolver estratégias de venda, e para o surgimento de mercados econômicos;
- *Ecologia*: para modelar fenômenos ecológicos;
- *Evolução e aprendizado*: para estudar como o aprendizado individual e a evolução das espécies se afetam;
- *Sistemas sociais*: para estudar aspectos evolutivos de sistemas sociais, como a evolução do comportamento social em colônias de insetos e, de forma mais geral, a evolução da cooperação e comunicação em sistemas multi-agentes;
- *Controladores robóticos*: para realizar controle de tarefas desenvolvidas por equipamentos robóticos, como planejamento de trajetórias para robôs móveis, desvio de obstáculos ou controle de manipuladores mecânicos.

## 4 ALGORITMOS GENÉTICOS

### 4.1 – Histórico dos AG's

#### 4.1.1 – Bases Biológicas e Evolucionistas

Em seu trabalho Yepes (2002) declarou que em meados do século XIX, *Charles Darwin* (1809-1882) revolucionou todo o pensamento acerca da evolução da vida e de nossas origens, provocando a maior discussão que já houve a respeito de uma teoria científica. Em seus dois livros - *Sobre a Origem das Espécies por Meio da Seleção Natural* (1859), e *A Descendência do Homem e Seleção em Relação ao Sexo* (1871) - Darwin defendia que o homem, tal qual os outros seres vivos, é resultado da evolução.

Em seus estudos, concluiu que nem todos os organismos que nascem, sobrevivem ou (o que é mais importante) reproduzem-se. Os indivíduos com mais oportunidades de sobrevivência seriam aqueles com características mais apropriadas para enfrentar as condições ambientais. Esses indivíduos teriam maior probabilidade de reproduzir-se e deixar descendentes. Nessas condições as variações favoráveis tenderiam a ser preservadas e as desfavoráveis, destruídas.

É um lento e constante processo de seleção ao longo das gerações, através do qual as espécies podem se diversificar, tornando-se mais adaptadas ao ambiente em que vivem.

Nascia assim o conceito de Seleção Natural, a grande contribuição de Darwin à teoria da evolução. Esse mecanismo de evolução pode ser resumido em seis etapas (MIT1996, apud YEPES, 2002):

- Os indivíduos de uma mesma espécie mostram muitas variações na forma e na fisiologia;
- Boa parte dessas variações é transmitida aos descendentes;
- Se todos os indivíduos de uma espécie se reproduzissem, as populações cresceriam exponencialmente;

- Como os recursos naturais são limitados, os indivíduos de uma população lutam por sua sobrevivência e de sua prole;
- Portanto, somente alguns (os mais *aptos*), sobrevivem e deixam descendentes. A sobrevivência e a reprodução diferencial dependem das características desses indivíduos que, por serem hereditárias, serão transmitidas aos seus filhos;
- Através dessa seleção natural, as espécies serão representadas por indivíduos cada vez mais adaptados.

O principal problema da Teoria Darwiniana foi a falta de uma teoria satisfatória que explicasse a origem e a transmissão das variações. Ele não conseguiu responder adequadamente a estas críticas e, somente com a posterior redescoberta das Leis de Mendel e das Mutações, foi que esses problemas puderam ser resolvidos.

Apesar disso, não há dúvida de que a teoria moderna da evolução deve mais a Darwin do que a qualquer outro cientista, e seu conceito de seleção natural continua válido até hoje.

#### **4.1.2 – Precursores dos Algoritmos Genéticos**

A primeira tentativa de representação, por meio de um modelo matemático, da teoria de Darwin, surgiu com o livro *The Genetic Theory of Natural Selection*, escrito pelo biólogo evolucionista Fisher.

A evolução era, tal como a aprendizagem, uma forma de adaptação, diferindo apenas na escala de tempo. Em vez de ser o processo de uma vida, era o processo de gerações. Como era feita em paralelo por um conjunto de organismos, tornava-se mais poderosa que a aprendizagem (YEPES, 2002).

A seguir, John Holland dedicou-se ao estudo de processos naturais adaptáveis, tendo inventado os Algoritmos Genéticos (AG's) em meados da década de 60. Ele os desenvolveu em conjunto com seus alunos e colegas da Universidade de Michigan nos anos 60 e 70, com o objetivo de estudar formalmente o fenômeno da adaptação como ocorre na natureza, e desenvolver modelos em que os mecanismos da adaptação natural pudessem ser importados

para os sistemas computacionais. Descobriu que é possível enfileirar uns e zeros ao longo de um pedaço de código de computador de maneira semelhante à forma como os genes são enfileirados em um cromossomo (LAUDON, 1999).

Como resultado do seu trabalho, em 1975, Holland edita *Adaptation in Natural and Artificial Systems* e, em 1989, David Goldberg edita *Genetic Algorithms in Search, Optimization and Machine Learning*, hoje considerados os livros mais importantes sobre AG's.

Os computadores podem combinar séries de uns e zeros da mesma forma que os cromossomos nos organismos vivos se combinam e recombinam, levando sucessivamente a melhores soluções. Os AG's proporcionam métodos de buscar todas as combinações de dígitos para identificar uma série (*string*) que represente a melhor solução para o problema (LAUDON, 1999).

Desde então, estes algoritmos vêm sendo aplicados com sucesso para problemas complexos que possuem um grande número de variáveis e cálculos, nos quais numerosas soluções alternativas devem ser testadas e nos mais diversos problemas de otimização e aprendizado de máquinas (POZO, 2000).

## **4.2 – Definições e Conceitos**

Algoritmos Genéticos são técnicas de busca baseadas nas Teorias da Evolução, nos quais as variáveis são representadas como genes em um cromossomo (indivíduo). Combinam a sobrevivência dos mais aptos com a troca de informação de uma forma estruturada, mas aleatória. O AG apresenta um grupo de soluções candidatas (população) na região de soluções. Por seleção natural e operadores genéticos, mutação e cruzamento, os cromossomos com melhor aptidão são encontrados. A seleção natural garante que os cromossomos mais aptos gerem descendentes nas populações futuras (YEPES, 2002).

Segundo Pozo (2000) esses algoritmos simulam processos naturais de sobrevivência e reprodução das populações, essenciais em sua evolução. Na natureza, indivíduos de uma mesma população competem entre si, buscando principalmente a sobrevivência, seja através da busca de recursos como alimento, ou visando a reprodução.

Os indivíduos mais aptos terão um maior número de descendentes, ao contrário dos indivíduos menos aptos.

Os requisitos para a implementação de um AG são:

- Representações das possíveis soluções do problema no formato de um código genético;
- População inicial que contenha diversidade suficiente para permitir ao algoritmo combinar características e produzir novas soluções;
- Existência de um método para medir a qualidade de uma solução potencial;
- Um procedimento de combinação de soluções para gerar novos indivíduos na população;
- Um critério de escolha das soluções que permanecerão na população ou que serão retirados desta;
- Um procedimento para introduzir periodicamente alterações em algumas soluções da população. Desse modo mantém-se a diversidade da população e a possibilidade de se produzir soluções inovadoras para serem avaliadas pelo critério de seleção dos mais aptos.

A idéia básica de funcionamento dos algoritmos genéticos é a de tratar as possíveis soluções do problema como "indivíduos" de uma "população", que irá "evoluir" a cada iteração ou "geração". Para isso é necessário construir um modelo de evolução onde os indivíduos sejam soluções de um problema. A execução do algoritmo pode ser resumida nos seguintes passos:

- Inicialmente escolhe-se uma população inicial, normalmente formada por indivíduos criados aleatoriamente;
- Avalia-se toda a população de indivíduos segundo algum critério, determinado por uma função que avalia a qualidade do indivíduo (função de aptidão ou "*fitness*");

Na prática podemos implementar facilmente um AG com o simples uso de strings de bits ou caracteres para representar os cromossomos e, com simples operações de manipulação de bits podemos implementar cruzamento, mutação e outros operadores genéticos.

O pseudocódigo de um AG básico é mostrado na figura 4. Nele, podemos ver que os AG's começam com uma população de  $n$  estruturas aleatórias (indivíduos), onde cada estrutura codifica uma solução do problema. O desempenho de cada indivíduo é avaliado com base numa função de avaliação de aptidão. Os melhores tenderão a ser os progenitores da geração seguinte, melhorando, de geração para geração, através da troca de informação (YEPES, 2002).

Algorithm AG is

t := 0;	determinar um tempo inicial
InitPop P(t);	iniciar uma população de indivíduos randômica
eval P(t);	avaliar aptidão dos indivíduos da população
while not done do	testar critérios (tempo, aptidão, etc.)
t := t + 1;	incrementar o contador de tempo
P' := SelectPar P(t);	selecionar os pares para cruzamento
Recombine P' (t);	realizar cruzamento dos pares selecionados
Mutate P' (t);	perturbar o grupo gerado pelo cruzamento
Evaluate P' (t);	avaliar as novas aptidões
P := Survive P, P'(t);	selecionar os sobreviventes
Od	
End AG.	

Figura 4: Pseudocódigo de um AG básico.

Fonte: YEPES, 2002.

Nota-se que a principal inovação dos AG's, no que se refere aos métodos de busca, é a implementação de um mecanismo de seleção de soluções no qual, em curto prazo os melhores têm maior probabilidade de sobreviver e, em longo prazo, os melhores têm maior probabilidade de ter descendência.

A vantagem principal dos AG's ao trabalharem com o conceito de população, ao contrário de muitos outros métodos que trabalham com um só ponto, é que eles encontram segurança na quantidade. Tendo uma população de pontos bem adaptados, é reduzida a possibilidade de alcançar um falso ótimo.

Os AG's podem ser aplicados praticamente em qualquer problema. Apesar de aleatórios, ou seja, suas buscas não são totalmente sem rumo, pois exploram informações históricas para encontrar novos pontos de busca onde são esperados melhores desempenhos. Isto é feito através de processos iterativos, ou seja, através de gerações.

Em Pozo (2002) descreve-se que em seguida, através do operador de seleção, escolhem-se os indivíduos de melhor valor (dado pela função de aptidão) como base para a criação de um novo conjunto de possíveis soluções, chamado de nova geração.

Esta nova geração é obtida aplicando-se sobre os indivíduos selecionados, operações que misturem suas características (chamadas genes), através dos operadores de cruzamento (*crossover*) e mutação;

Estes passos são repetidos até que uma solução aceitável seja encontrada, até que o número predeterminado de passos seja atingido ou até que o algoritmo não consiga mais melhorar a solução já encontrada.

Os principais componentes mostrados na são descritos a seguir em mais detalhes.

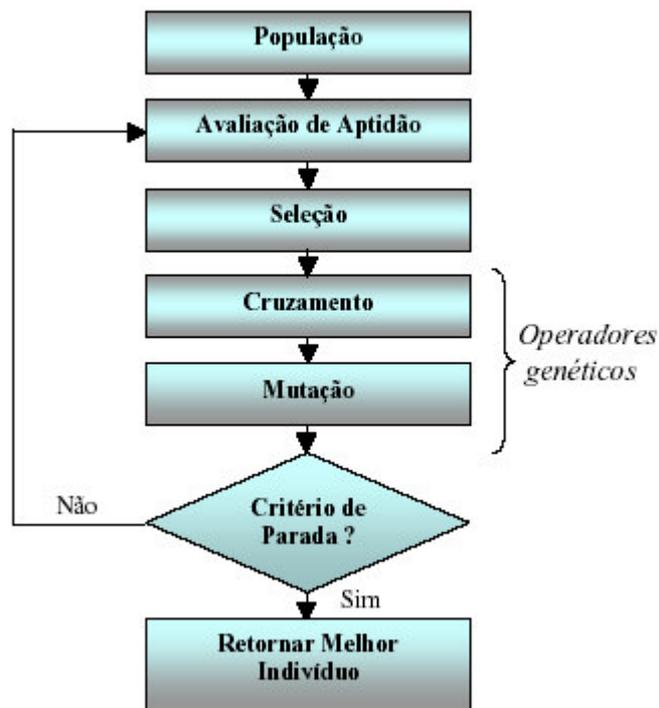


Figura 5: Estrutura Básica de um AG  
Fonte: POZO, 2000.

### **4.2.1 - População**

A população de um algoritmo genético é o conjunto de indivíduos que estão sendo cogitados como solução e que serão usados para criar o novo conjunto de indivíduos para análise. O tamanho da população pode afetar o desempenho global e a eficiência dos algoritmos genéticos. Populações muito pequenas têm grandes chances de perder a diversidade necessária para convergir a uma boa solução, pois fornecem uma pequena cobertura do espaço de busca do problema. Entretanto, se a população tiver muitos indivíduos, o algoritmo poderá perder grande parte de sua eficiência pela demora em avaliar a função de aptidão de todo o conjunto a cada iteração, além de ser necessário trabalhar com maiores recursos computacionais.

O ponto de partida para a utilização de um algoritmo genético como ferramenta para solução de problemas é a representação destes problemas de maneira que os algoritmos genéticos possam trabalhar adequadamente sobre eles. Uma das principais formas é representar cada atributo como uma seqüência de bits e o indivíduo como a concatenação das seqüências de bits de todos os seus atributos.

A codificação usando o próprio alfabeto do atributo que se quer representar (letras, códigos, números reais, etc.) para representar um indivíduo também é muito utilizada.

Diversas outras formas são possíveis, normalmente a forma mais apropriada está fortemente ligada ao tipo de problema.

### **4.2.2 - Avaliação de Aptidão (Fitness)**

Neste componente será calculado, através de uma determinada função, o valor de aptidão de cada indivíduo da população. Este é o componente mais importante de qualquer algoritmo genético. É através desta função que se mede quão próximo um indivíduo está da solução desejada ou quão boa é esta solução.

É essencial que esta função seja muito representativa e diferencie na proporção correta as más soluções das boas. Se houver pouca precisão na avaliação, uma ótima solução pode ser posta de lado durante a execução do algoritmo, além de gastar mais tempo explorando soluções pouco promissoras.

### 4.2.3 - Seleção

Dada uma população em que a cada indivíduo foi atribuído um valor de aptidão, existem vários métodos para selecionar os indivíduos sobre os quais serão aplicados os operadores genéticos. Há diversas formas de seleção, entre eles há o método de seleção por Roleta e o método de seleção por Torneio.

No método de seleção por Roleta (figura 6), cada indivíduo da população é representado na roleta proporcionalmente ao seu índice de aptidão. Assim, para indivíduos com alta aptidão é dada uma porção maior da roleta, enquanto aos indivíduos de aptidão mais baixa, é dada uma porção relativamente menor.

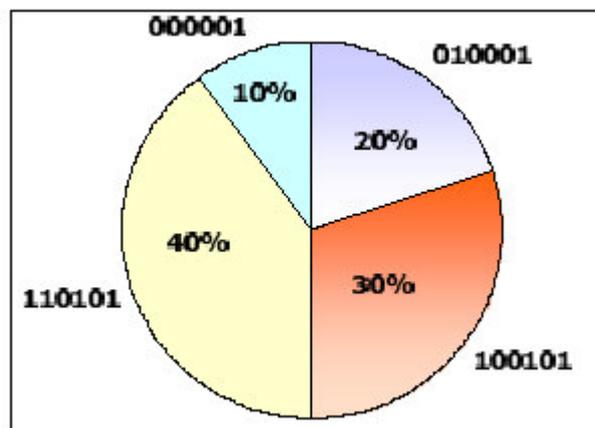


Figura 6: Método de Seleção por Roleta

Fonte: POZO, 2000.

Neste método, um dos problemas encontrados pode ser o tempo de processamento, já que o método exige duas passagens por todos os indivíduos da população.

Um exemplo da implementação deste método, segundo Mitchell (1997 apud POZO, 2000) é mostrado na figura 7:

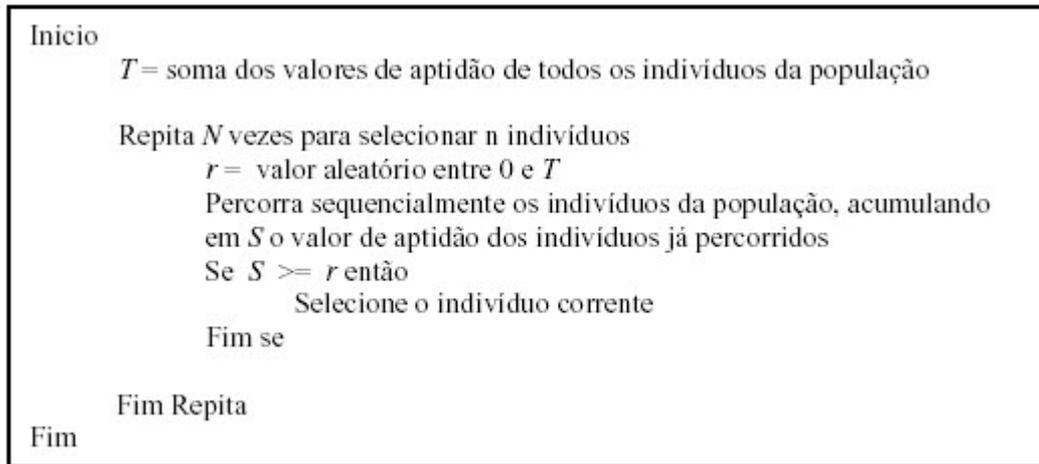


Figura 7: Algoritmo Básico do Método de Seleção por Roleta

Um outro método é a seleção por Torneio, onde um número  $n$  de indivíduos da população é escolhido aleatoriamente para formar uma sub-população temporária. Deste grupo, o indivíduo selecionado dependerá de uma probabilidade  $k$  definida previamente.

Um exemplo básico da implementação deste algoritmo Mitchell (1997 apud POZO, 2000) é mostrado na figura 8:

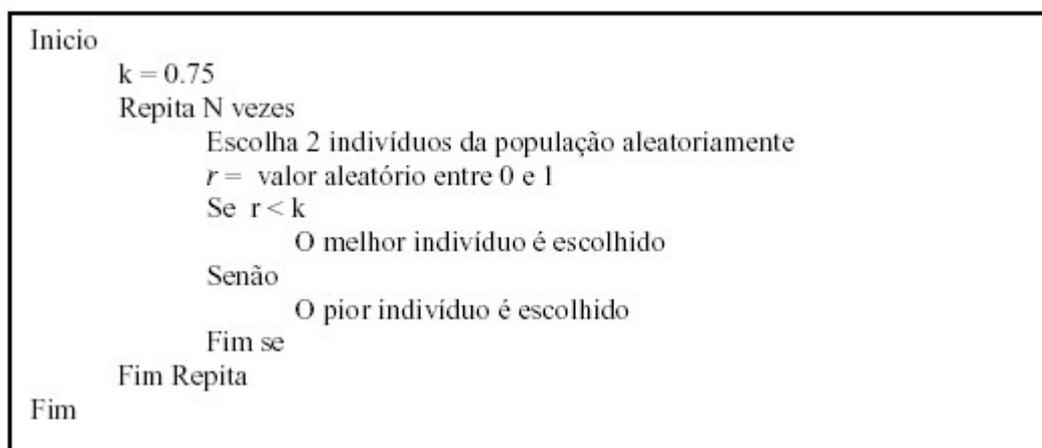


Figura 8: Algoritmo Básico do Método de Seleção por Torneio

Este método é o mais utilizado, pois oferece a vantagem de não exigir que a comparação seja feita entre todos os indivíduos da população.

### 4.3 - Operadores Genéticos

O princípio básico dos operadores genéticos é transformar a população através de sucessivas gerações, estendendo a busca até chegar a um resultado satisfatório. Os operadores genéticos são necessários para que a população se diversifique e mantenha características de adaptação adquiridas pelas gerações anteriores. Os operadores de cruzamento e de mutação têm um papel fundamental em um algoritmo genético.

#### 4.3.1 - Cruzamento (Crossover)

Este é considerado o operador genético predominante. Através do cruzamento são criados novos indivíduos misturando características de dois indivíduos pais. Esta mistura é feita tentando imitar (em um alto nível de abstração) a reprodução de genes em células. Trechos das características de um indivíduo são trocados pelo trecho equivalente do outro. O resultado desta operação é um indivíduo que potencialmente combine as melhores características dos indivíduos usados como base.

Alguns tipos de cruzamento bastante utilizados são o cruzamento em um ponto e o cruzamento em dois pontos.

Com um ponto de cruzamento, seleciona-se aleatoriamente um ponto de corte do cromossomo. Cada um dos dois descendentes recebe informação genética de cada um dos pais.

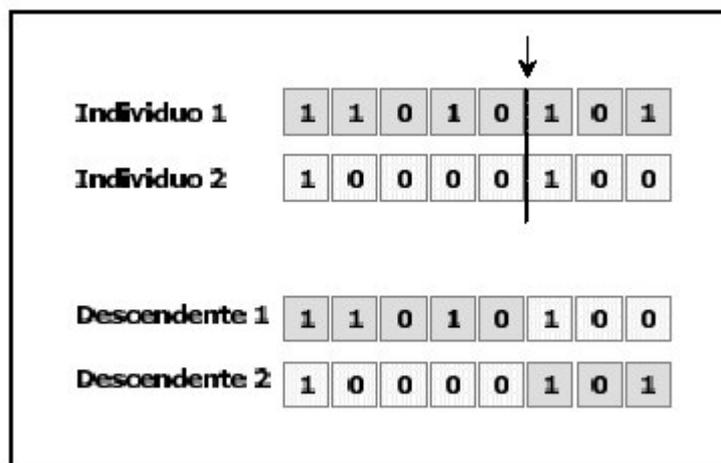


Figura 9: Cruzamento em Um Ponto  
Fonte: POZO, 2000.

Com dois pontos de cruzamento, um dos descendentes fica com a parte central de um dos pais e as partes extremas do outro pai e vice versa.

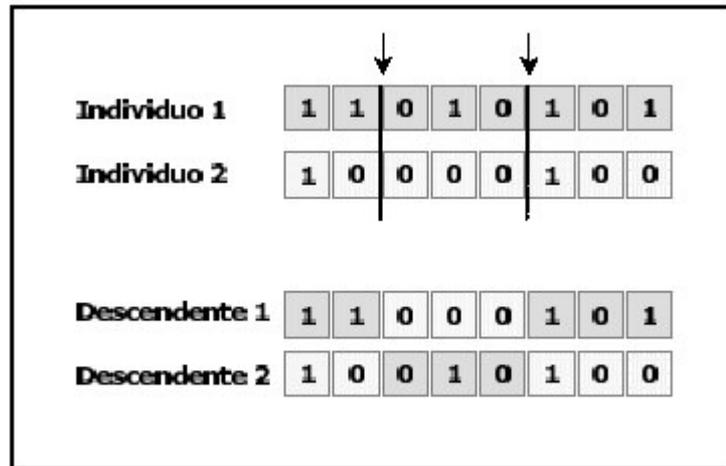


Figura 10: Cruzamento em Dois Pontos  
Fonte: POZO,2000.

#### 4.3.2 - Mutação

Esta operação simplesmente modifica aleatoriamente alguma característica do indivíduo sobre o qual é aplicada. Esta troca é importante, pois acaba por criar novos valores de características que não existiam ou apareciam em pequena quantidade na população em análise.

O operador de mutação é necessário para a introdução e manutenção da diversidade genética da população. Desta forma, a mutação assegura que a probabilidade de se chegar a qualquer ponto do espaço de busca possivelmente não será zero. O operador de mutação é aplicado aos indivíduos através de uma taxa de mutação geralmente pequena como mostrado abaixo:

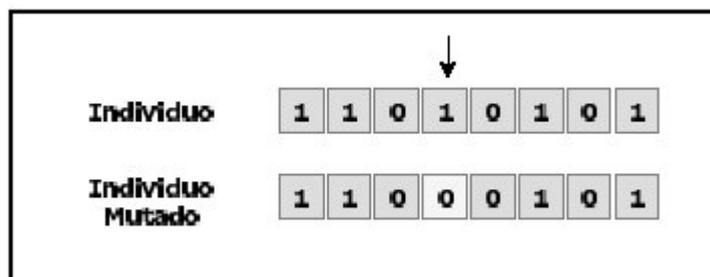


Figura 11: Mutação Simples

Fonte: POZO, 2000.

#### 4.4 – Parâmetros Genéticos

É importante analisar de que maneira alguns parâmetros influem no comportamento dos Algoritmos Genéticos, para que se possa estabelecê-los conforme as necessidades do problema e dos recursos disponíveis. A seguir serão listados alguns Parâmetros Genéticos utilizados (RIO, 1992 apud YEPES, 2002):

- *Tamanho da População:* O tamanho da população determina o número de cromossomos na população, afetando diretamente o desempenho global e a eficiência dos AG's. Com uma população pequena o desempenho pode cair, pois deste modo, a população fornece uma pequena cobertura do espaço de busca do problema. Uma grande população geralmente fornece uma cobertura representativa do domínio do problema, além de prevenir convergências prematuras para soluções locais ao invés de globais.
- *Taxa de Cruzamento:* Quanto maior for esta taxa, mais rapidamente novas estruturas serão introduzidas na população. Mas se esta for muito alta, a maior parte da população será substituída, e pode ocorrer perda de estruturas de alta aptidão. Com um valor baixo, o algoritmo pode tornar-se muito lento;
- *Tipo de Cruzamento:* O tipo de cruzamento a ser utilizado determina a forma como se procederá a troca de segmentos de informação entre os casais de cromossomos selecionados para cruzamento. O ideal é testar diversos tipos de cruzamento em conjunto com as outras configurações do AG em uso, para verificar qual apresenta um melhor resultado;
- *Taxa de Mutação:* Determina a probabilidade em que uma mutação ocorrerá. Mutação é utilizada para dar nova informação para a população e também para prevenir que a população se sature com cromossomos semelhantes (Convergência Prematura). Uma baixa taxa de mutação previne que uma dada posição fique estagnada em um valor, além de possibilitar que se chegue em qualquer ponto do espaço de busca. Com uma taxa muito alta a busca se torna essencialmente aleatória além de aumentar muito a possibilidade de que uma boa solução seja destruída. A melhor Taxa de Mutação é dependente da aplicação, mas, para a maioria dos casos é entre 0,001 e 0,1.

## **4.5 - Geração**

A cada passo, um novo conjunto de indivíduos é gerado a partir da população anterior. A este novo conjunto dá-se o nome de Geração. É através da criação de uma grande quantidade de gerações que é possível obter resultados dos Algoritmos Genéticos.

## 5 IMPLANTAÇÃO, PROGRAMAÇÃO E USABILIDADE DE ALGORITMOS GENÉTICOS

### 5.1 – Métodos e Critérios para Implantação de um AG

Para implantar um AG, necessita-se definir de forma correta os seguintes métodos e critérios (SER, 1996 apud YEPES, 2002):

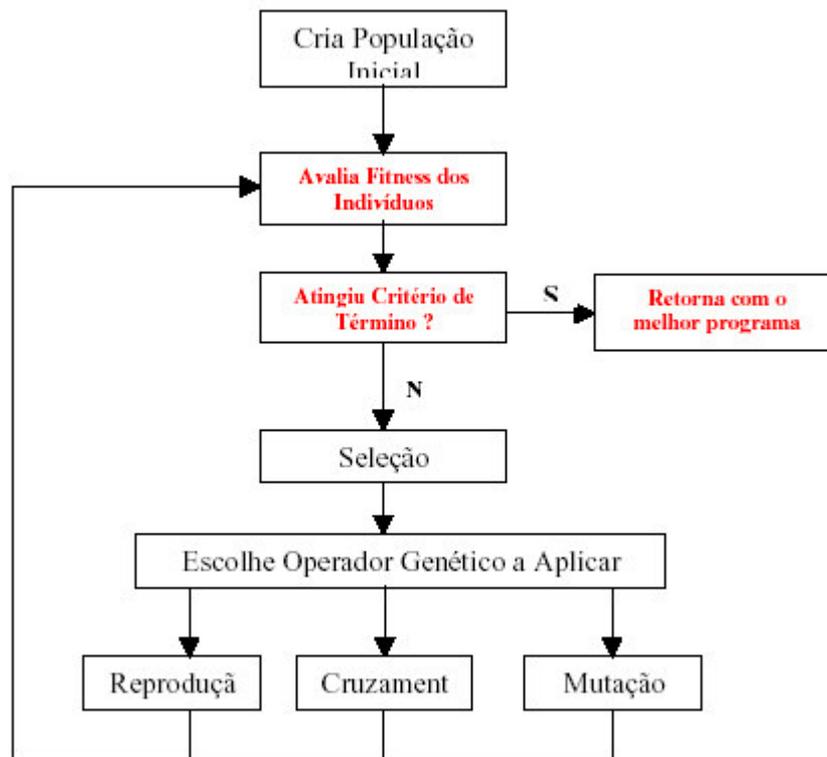


Figura 12: Estrutura do Algoritmo de Programação Genética

Fonte: YEPES, 2002.

- *Critério de Inicialização*: refere-se a como deve ser construída a população inicial, com a qual se inicializará o AG;
- *Funções de Avaliação e Aptidão*: deve ser determinada a função de avaliação mais apropriada para o problema, assim como a função de aptidão que utilizará o AG para resolvê-lo;

- *Critério de Parada:* devem ser determinadas as condições nas quais se considera que o AG encontrou uma solução aceitável ou tenha fracassado no processo de busca e não faça sentido continuar;
- *Crítérios de Seleção:* a seleção deve dirigir o processo de busca em favor dos indivíduos mais aptos. Isto pode ser feito de varias maneiras como, por exemplo, por amostragem direta, por amostragem aleatória simples ou por amostragem estocástica;
- *Operadores Genéticos:* denominação dada aos operadores utilizados para levar a cabo a reprodução. Todo AG faz uso de pelo menos três Operadores Genéticos (levando em conta o AG básico): *seleção, cruzamento e mutação.*
- *Crítério de Substituição:* os critérios com que se selecionam os criadores não necessariamente têm que ser os mesmos usados para selecionar os sobreviventes, por isso há necessidade de especificá-los separadamente, são eles:
  - *Substituição Imediata:* os descendentes substituem seus progenitores;
  - *Substituição por Fator Cheio:* os descendentes substituem aqueles membros da população mais parecidos com eles;
  - *Substituição por Inserção:* são selecionados para serem eliminados membros da população de criadores (geralmente os piores), aos quais serão substituídos pelos descendentes;
  - *Substituição por Inclusão:* os descendentes são somados aos n progenitores em uma única população e nesta, são selecionados os n melhores.
- *Parâmetros de Funcionamento:* um AG precisa que lhe proporcionem certos parâmetros de funcionamento como tamanho da população, probabilidades de aplicação dos Operadores Genéticos, precisão da codificação entre outros.

A seleção de um ou outro método ou critério acima depende não apenas do tipo de problema a ser resolvido, mas também de que não sejam violados certos requisitos imprescindíveis ao bom funcionamento do AG.

## 5.2 - Operadores Genéticos

Uma vez que os indivíduos tenham sido selecionados, deve-se aplicar um dos operadores genéticos. Os três operadores principais são (Koza, 1992 APUD Pozo, 2000):

- **Reprodução:** um programa é selecionado e copiado para a próxima geração sem sofrer nenhuma mudança em sua estrutura;
- **Cruzamento** (*crossover*): dois programas são selecionados e são recombinados para gerar outros dois programas. Um ponto aleatório de cruzamento é escolhido em cada programa-pai e as árvores abaixo destes pontos são trocadas.

Para que o cruzamento seja sempre possível, o conjunto de funções deve apresentar a propriedade de Fechamento (*closure*), isto é, as funções devem suportar como argumento qualquer outra função ou terminal. Se não for possível, deve-se estabelecer critérios de restrição na escolha dos pontos de cruzamento;

- **Mutação** (*mutation*): um programa é selecionado e um de seus nós é escolhido aleatoriamente. A árvore cuja raiz é o nó selecionado é então eliminada e substituída por uma nova árvore gerada aleatoriamente.

## 5.3 – Critério de Término

É responsável por interromper o laço de repetição do processo evolutivo que, idealmente, não teria fim. O critério mais comum é limitar o número máximo de gerações ou até que uma solução satisfatória seja encontrada, porém existem critérios baseados no próprio acompanhamento do processo evolutivo, isto é, enquanto houver melhoria na média da população, o processo evolutivo prossegue.

## 5.4 - Limitações

A obrigatoriedade da propriedade de fechamento (*closure*) limita os domínios a serem usados, não possibilitando a aplicação ampla da Programação Genética. A necessidade do fechamento é devida ao uso irrestrito dos operadores genéticos nos programas.

## 6 O PROBLEMA DO CAIXEIRO VIAJANTE

O Problema do Caixeiro Viajante (PCV) ou *Travelling Salesman Problem (TSP)* é um dos mais tradicionais e conhecidos problemas de programação matemática.

O objetivo do PCV é encontrar um roteiro e uma distância cujo comprimento e tempo total sejam os menores possíveis em um dado conjunto de cidades (BOAVENTURA, 2001).

Sua representação é feita através de um grafo que em uma definição bem simples, é um conjunto de vértices e arestas. Os vértices (ou nós) são pontos que podem representar cidades e as arestas são linhas que conectam os vértices, podendo representar ruas ou estradas. É um passeio que percorre todos os vértices de um grafo e retorna ao vértice origem (início do passeio), passando por cada vértice apenas uma vez.

O Problema do Caixeiro Viajante é importante devido a pelo menos três de suas características: grande aplicação prática, grande relação com outros modelos e grande dificuldade de solução exata. Em suas diversas versões, o PCV está presente em inúmeros problemas práticos, como por exemplo, (BOAVENTURA, 2001):

- Programação de operações em máquinas;
- Otimização do movimento de ferramentas de corte;
- Otimização de perfurações de furos em placas de circuitos impressos;
- Na solução de problemas de seqüenciamento de código genético;
- Na solução de problemas de programação e distribuição de tarefas;
- Trabalhos administrativos;
- E nesse trabalho, para o Problema de Roteamento ou planejamento dos Veículos (PRV) que fazem a coleta e distribuição de correspondências nas cidades da Zona da Mata mineira.

Porém, o PCV pertence à classe de problemas considerados difíceis ou intratáveis, devido a sua complexidade.

A busca de uma solução ótima pode "custar caro", pois a medida que aumentamos o número de vértices o tempo para a resolução do problema aumenta de forma exponencial, tornando as vezes a resolução inviável para os computadores atuais (BOAVENTURA, 2001).

Para resolver problemas difíceis, algumas técnicas são utilizadas, como por exemplo, os Algoritmos de aproximação que apresentam uma solução correta com a garantia de que esteja dentro de uma determinada porcentagem da solução ótima. Fazendo uma análise do pior e do melhor caso que o algoritmo pode produzir, é possível avaliar sua complexidade e a proximidade das suas soluções em relação aquela que é ótima.

Com o uso de algoritmos de aproximação é possível encontrar, para problemas difíceis, soluções de boa qualidade em tempo computacional aceitável, devido a isso, a otimização do problema de roteamento de veículos da logística dos correios será baseada na lógica de funcionamento dos Algoritmos Genéticos e do Sistema GIDEON.

## 7 LOGÍSTICA DOS CORREIOS

### 7.1 – Configuração do Encaminhamento Postal da Região da Zona da Mata Mineira

O encaminhamento postal da região da Zona da Mata, abaixo ilustrado, foi enviado pelo Gerente de Atividades Externas Antônio Carlos Vilhena do CTCE de Juiz de Fora -MG, onde:

- LTR: Linha Tronco Regional;
- LCE: Linha De Coleta e Entrega.

As demais cidades recebem a carga através da “LA” – Linhas Auxiliares, que são transportadas através das empresas de ônibus que fazem as linhas regulares de transportes de passageiros.

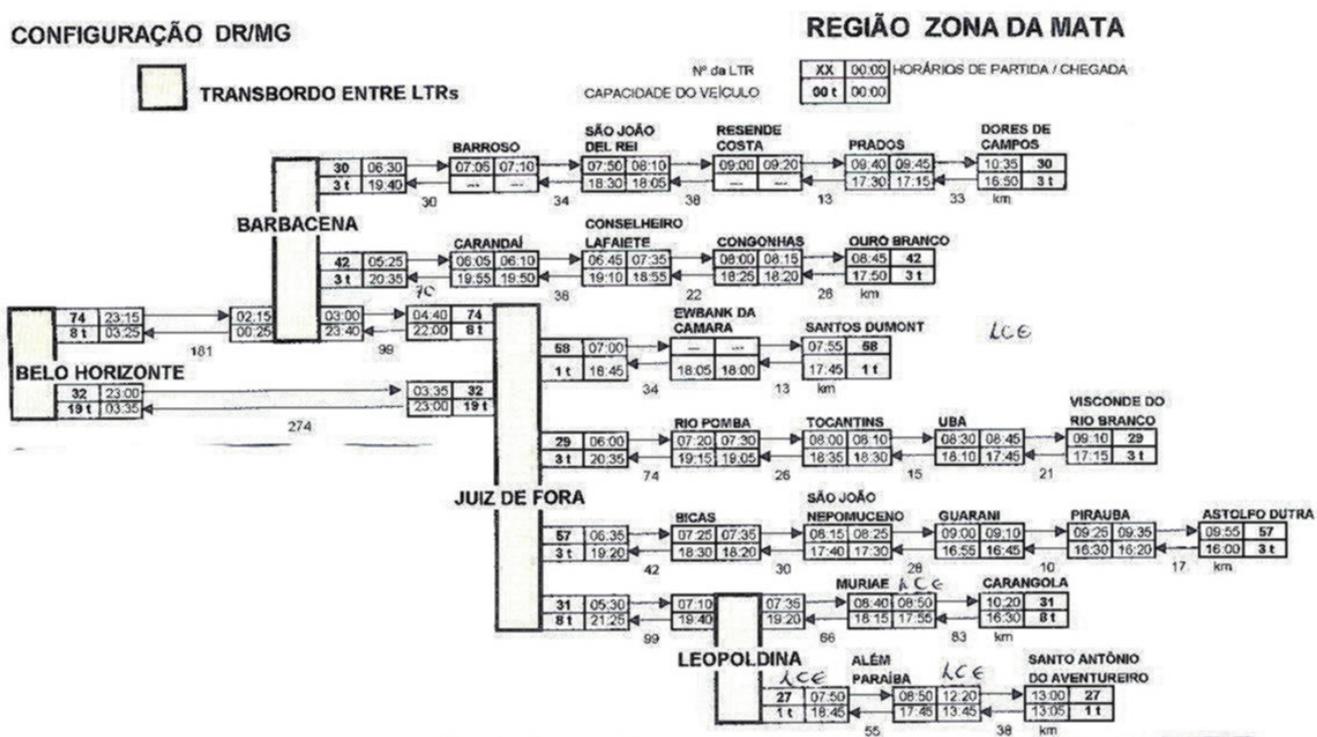


Figura 13: Configuração do Encaminhamento Postal da Região da Zona da Mata Mineira

## 8 SISTEMA GIDEON

Thangiah (1991 apud ARAKAKI, 1998) na 7ª conferência de Aplicação de Inteligência Artificial em Miami, propôs um sistema de Algoritmo Genético para resolver o Problema de Roteamento de Veículos.

Seu sistema chamado de GIDEON consiste de dois módulos, um módulo de agrupamento global que atribui clientes aos veículos por um processo chamado setorização genética e outro módulo que faz uma otimização local das rotas.

Inicialmente, os clientes são divididos em setores ou agrupamentos utilizando um algoritmo de varredura. A cada cliente é atribuído um pseudo-ângulo em coordenadas adjacentes. Os clientes são divididos em  $K$  agrupamentos ( $K$  é o tamanho da frota de veículos) baseados num conjunto de ângulos “sementes”. A Cada cliente é atribuído a um determinado agrupamento de acordo com seu ângulo como mostra a figura abaixo:

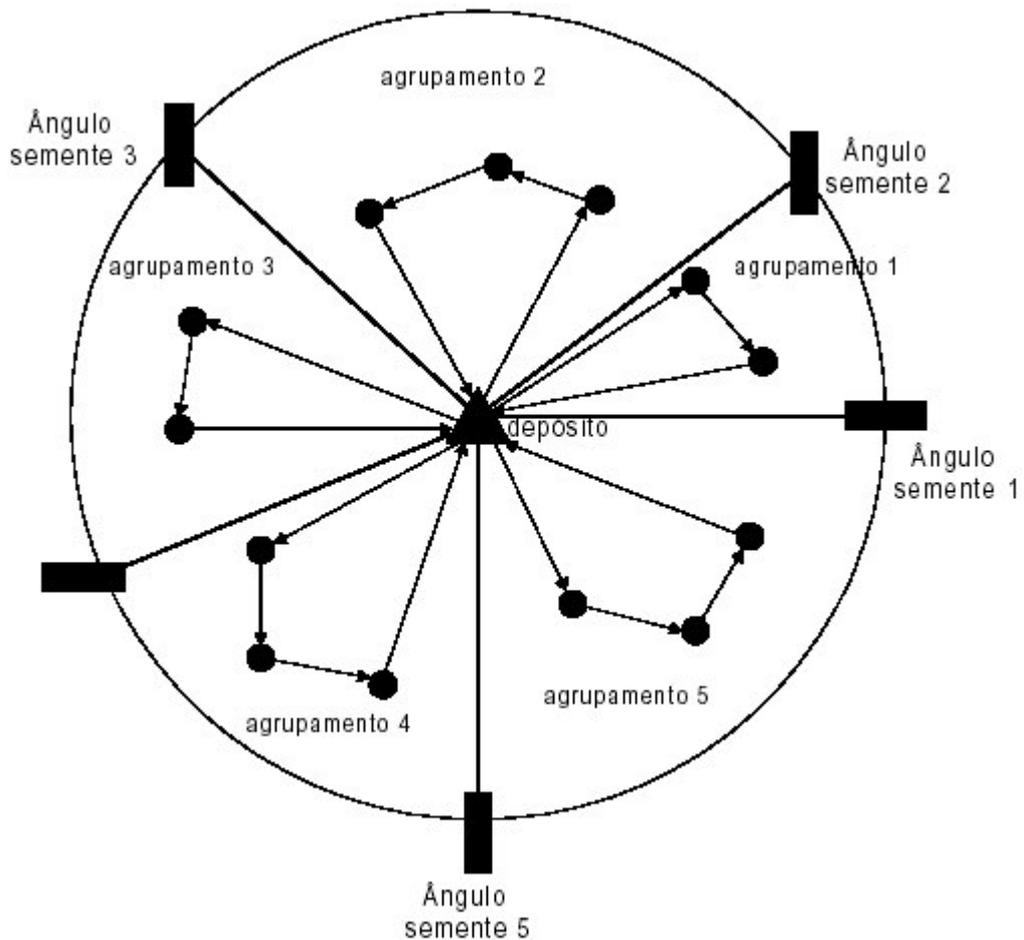


Figura 14: Divisão de clientes utilizando ângulos sementes  
Fonte: ARAKAKI, 1998.

O AG é utilizado para encontrar a melhor configuração de ângulos sementes.

Durante o processo de setorização genética cada cromossomo representa um conjunto de ângulos sementes. Cada ângulo é representado por 3 bits e para um problema de K veículos. K-1 ângulos sementes são necessários (o primeiro ângulo semente é igual a 0), portanto o comprimento de cada cromossomo será  $(K-1)*3$ . A população inicial é gerada aleatoriamente.

Uma conversão de bits para ângulo semente é realizada. A função de avaliação é o custo total das rotas dos agrupamentos formados a partir do conjunto de ângulos sementes derivados do cromossomo.

Foram utilizados operadores de cruzamento tradicionais (2-pontos). Portanto, os clientes são particionados de acordo com os ângulos sementes e as rotas são determinadas por uma heurística de inserção do mais barato. O módulo de otimização de rota procura melhorar a solução pela movimentação e troca de clientes dentro e entre os agrupamentos.

Quatro tipos de operações são realizados pelo módulo: mover um cliente, mover dois clientes, trocar um cliente e trocar dois clientes e em cada uma delas é avaliado o custo da rota. Estas operações de otimização local são realizadas até que não haja mais redução de custo, sendo que esta nova solução pode ser introduzida novamente no primeiro módulo.

Com isso Thangiah (1991 apud ARAKAKI, 1998) verificou que duas iterações são suficientes para obter boas soluções. O sistema GIDEON apresentou uma redução média de 3,9% no tamanho da frota e 4,4% na distância percorrida pelos veículos para 56 problemas.

## **9 SIMULAÇÃO DAS APLICAÇÕES DE ALGORITMOS GENÉTICOS PARA OTIMIZAÇÃO DA LOGÍSTICA DOS CORREIOS NA ZONA DA MATA MINEIRA, UTILIZANDO O PROBLEMA DO CAIXEIRO VIAJANTE E A LÓGICA DE FUNCIONAMENTO DO SISTEMA GIDEON**

A simulação desta aplicação baseia-se na monografia apresentada para o Exame de Qualificação do Curso de Computação Aplicada (CAP), no Instituto Nacional de Pesquisas Espaciais (INPE) por Reinaldo Gen Ichiro Arakaki em 1998, para o problema de roteamento de veículos utilizando algumas metaheurísticas.

### **9.1 – Descrição do Problema**

As correspondências são coletadas e entregues a partir de um pólo central representado pela cidade de Belo Horizonte, por uma frota de veículos homogêneos.

Cada veículo realiza um percurso saindo do pólo central, para um subconjunto de cidades representados por Juiz de Fora e Barbacena, que repassam posteriormente a seus agregados.

Porém, a rota de cada veículo deve obedecer a algumas restrições como: a capacidade do veículo e o tempo limite para realizar uma rota.

Este problema de roteamento pretende traçar as melhores rotas para distribuição e coleta, sem violar as restrições e otimizando a função objetivo. Normalmente são consideradas três funções objetivo (ARAKAKI, 1998):

- Minimizar a distância total percorrida e tempo gasto por todos os veículos;
- Minimizar o número de veículos e deste número mínimo, minimizar a distância total percorrida;
- Minimizar a combinação de custo de veículos e distância percorrida.

Pra exemplificar, é considerado um conjunto de treze cidades ordenadas radicalmente, onde os valores entre parênteses são as distâncias em quilômetros entre cada uma e a capacidade média dos veículos de aproximadamente 5,4 toneladas.

A figura 15 apresenta a solução viável e utiliza 12 veículos. A melhor solução utiliza 10 veículos e está representada pela figura 16

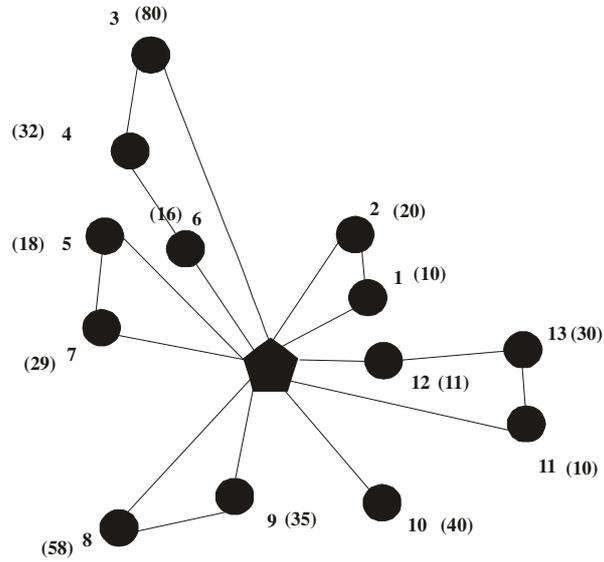


Figura 15: Solução Viável

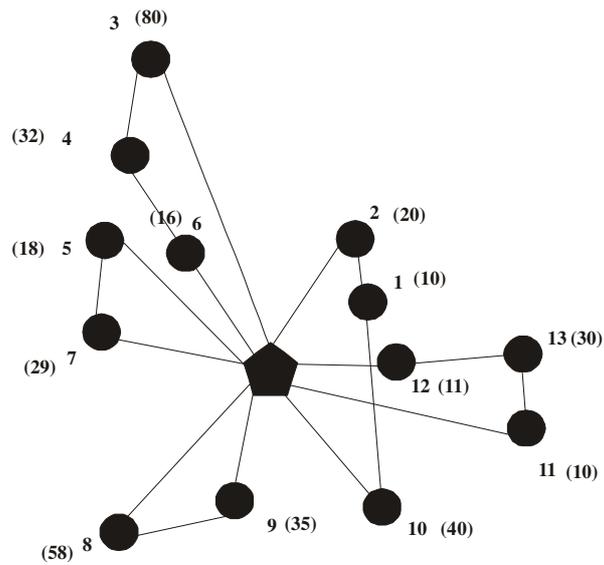


Figura 16: Melhor Solução

Uma formulação de programação matemática para o PRV foi dada por Bodin (1983 apud ARAKAKI, 1998).

A formulação considera o pólo central como nó 0 e as subcidades são numeradas de 1 a n, conforme mostra abaixo (ARAKAKI, 1998):

$$\text{Minimize } \sum_{i=0}^n \sum_{j=0}^n \sum_{v=1}^K c_{ij}^v x_{ij}^v \quad (1.1)$$

$$\sum_{i=0}^n \sum_{v=1}^K x_{ij}^v = 1 \quad j = 1, \dots, n \quad (1.2)$$

$$\sum_{i=0}^n \sum_{v=1}^K x_{ij}^v = 1 \quad i = 1, \dots, n \quad (1.3)$$

$$\sum_{i=0}^n x_{ip}^v - \sum_{j=0}^n x_{pj}^v = 0 \quad j = 1, \dots, n \quad (1.4)$$

$$\sum_{j=0}^n q_j \left( \sum_{i=0}^n x_{ij}^v \right) \leq Q_v \quad v = 1, \dots, K \quad (1.5)$$

$$\sum_{j=1}^n \delta_j^v \left( \sum_{i=0}^n x_{ij}^v \right) + \sum_{i=0}^n \sum_{j=0}^n t_{ij}^v x_{ij}^v \leq T_v \quad v = 1, \dots, K \quad (1.6)$$

$$\sum_{j=1}^n x_{0j}^v \leq 1 \quad v = 1, \dots, K \quad (1.7)$$

$$\sum_{j=1}^n x_{i0}^v \leq 1 \quad v = 1, \dots, K \quad (1.8)$$

$$X \in S \quad (1.9)$$

$$x_{ij}^v \in \{0,1\} \text{ para todo } i, j, k \quad (1.10)$$

Onde :

$n$  = número de cidades;

$K$  = número de veículos;

$C_{ij}^v$  = custo de viagem do nó  $i$  ao nó  $j$  para o veículo  $v$ ;

$X_{ij}^v = 1$ , se veículo viaja do nó  $i$  ao nó  $j$ ; e  $0$  caso contrário;

$q_i$  = quantidade de carga da cidade  $i$ ;

$Q_v$  = capacidade do veículo  $v$ ;

$\delta_i^v$  = tempo de descarregamento e carregamento da cidade  $i$  para o veículo  $v$ ;

$t_{ij}^v$  = tempo de viagem do nó  $i$  para o nó  $j$  e para o veículo  $v$ ;

$T_v$  = tempo de rota máximo permitido para o veículo  $v$ ;

$X$  = matriz de componentes  $x_{ij} \equiv \sum_{v=1}^k X_{ij}^v$

As restrições 1.2 e 1.3 asseguram que cada cidade é visitada exatamente uma vez. A continuidade da rota é garantida pela 1.4, onde se um veículo chega no ponto de entrega ou coleta deve também partir daquele ponto. A 1.5 é a restrição da capacidade do veículo. A 1.6 limita o máximo comprimento da rota. A 1.7 e 1.8 asseguram que cada veículo é usado somente uma vez e a restrição 1.9 previne que a rota de um veículo se desconecte.

## **9.2 – Algoritmos Genéticos aplicados ao Problema do Caixeiro Viajante na Logística dos Correios**

De acordo com a lógica do Sistema GIDEON de Thangiah (1991 apud ARAKAKI, 1998) faz-se a simulação para essa aplicação.

O Sistema consiste em dois módulos, portanto um módulo de agrupamento global que atribui cidades aos veículos por um processo chamado setorização genética e outro módulo que faz uma otimização local das rotas.

Inicialmente, as cidades são divididas em setores ou agrupamentos utilizando algum algoritmo de varredura. A cada cidade é atribuído um pseudo-ângulo em coordenadas polares adjacentes.

As cidades são divididos em  $K$  agrupamentos ( $K$  é o tamanho da frota de veículos) baseados num conjunto de ângulos “sementes”. A Cada cidade é atribuído um determinado agrupamento de acordo com seu ângulo como mostra a figura 17.

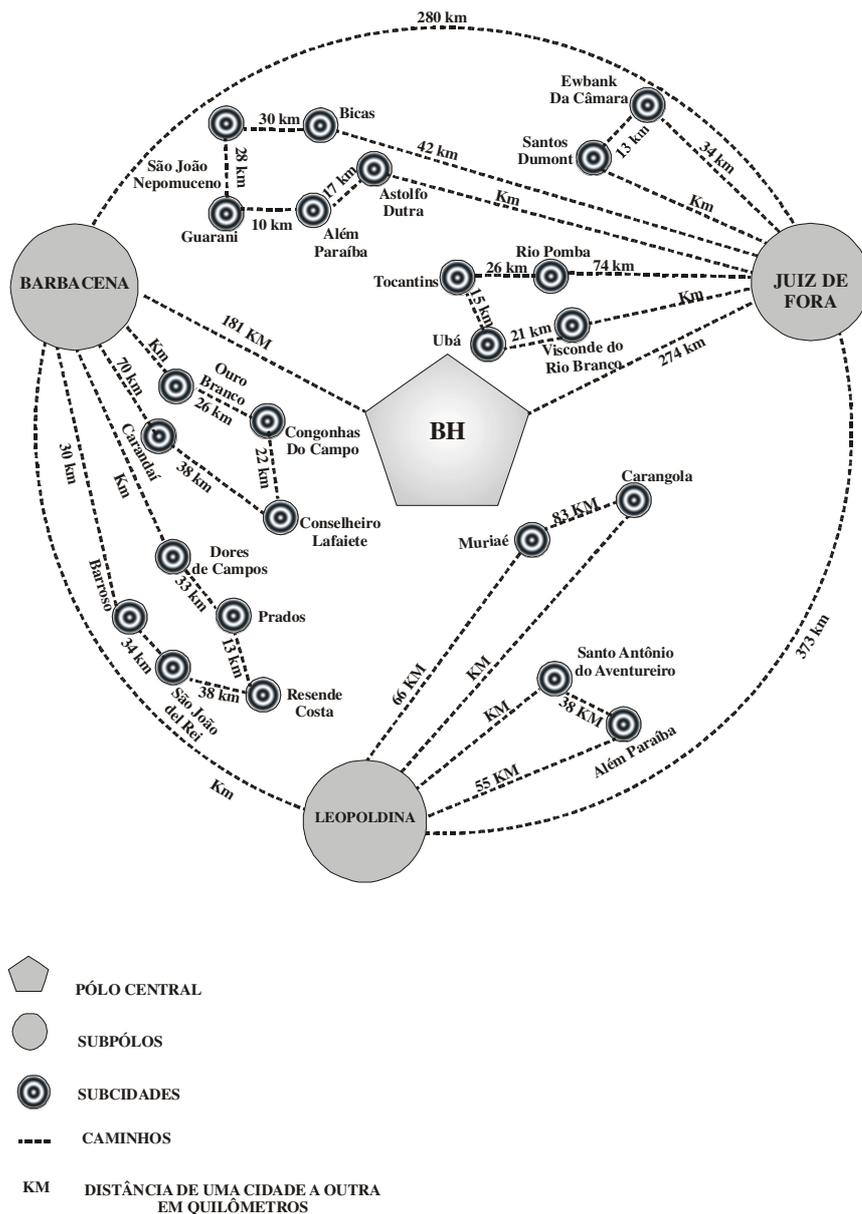


Figura 17 – Divisão da Configuração Postal da Região do Campo das Vertentes utilizando ângulos sementes

O AG é utilizado para encontrar a melhor configuração de ângulos sementes. Durante o processo de setorização genética cada cromossomo representa um conjunto de ângulos sementes. Cada ângulo semente é representado por N bits e para um problema de K veículos, K-1 ângulos sementes são necessários, sendo o primeiro ângulo semente é igual a 0. Portanto o comprimento de cada cromossomo será  $(K-1)*N$  e a população inicial é gerada aleatoriamente.

Uma conversão de bits para ângulo semente é realizada. A função de avaliação é o custo total das rotas dos agrupamentos formados a partir do conjunto de ângulos sementes derivados do cromossomo. Foram utilizados operadores de cruzamento tradicionais em 2-pontos propostos por Thangiah (1991 apud ARAKAKI, 1998).

Assim as cidades são particionadas de acordo com os ângulos sementes e as rotas são determinadas por uma heurística de inserção do mais barato.

O módulo de otimização de rota procura melhorar a solução pela movimentação e troca de cidades dentro e entre os agrupamentos.

Quatro tipos de operações são realizadas pelo módulo: mover de uma cidade a outra, mover de duas cidades, trocar de cidade e trocar mais cidades, onde em cada uma é avaliado o custo da rota.

Estas operações de otimização local são realizadas até que não haja mais redução de custo, podendo a nova solução ser introduzida novamente no primeiro módulo, para a obtenção de boas soluções.

Supondo que duas iterações sejam suficientes para obter boas soluções, para essa aplicação o sistema GIDEON Thangiah (1991 apud ARAKAKI, 1998), apresentaria uma redução média de 14% no tamanho da frota e 15,79% na distância percorrida pelos veículos para as 28 cidades que correspondem ao Encaminhamento Postal da Região da Zona da Mata Mineira.

## 10 CONCLUSÃO

O objetivo da inteligência Artificial e Computacional de implementar em computadores ações do pensamento humano ou do funcionamento cerebral, torna conveniente sua aplicação em determinadas categorias de problemas de difícil solução, devido sua alta complexidade matemática e computacional.

Porém o uso de técnicas baseadas na Evolução Biológica como os Algoritmos Genéticos, tem-se mostrado bastante promissor, quando utilizados para problemas complexos como a otimização em roteamentos, planejamentos e a busca de soluções aproximadamente ótimas.

Os AG's por trabalharem em duas etapas foram escolhidos como a heurística a ser utilizada, pois trabalham com uma busca tradicional para encontrar uma solução inicial viável e a partir disto utilizando cada uma delas, procuram melhorar a solução a partir do conceito de vizinhança.

A escolha da utilização do Sistema GIDEON de Thangiah (1991 apud ARAKAKI, 1998), surgiu devido aos bons resultados encontrados para o Problema de Roteamento de Veículos usando AG's.

Com a simulação de sua aplicação lógica, obtiveram-se resultados bastante interessantes, esses apresentariam uma redução média de 14% no tamanho da frota e 15,8% na distância percorrida pelos veículos para as 28 cidades que correspondem ao encaminhamento postal da região.

Através do estudo realizado percebeu-se o quão interessante é o problema de roteamento e planejamento e, porque existem diversos pesquisadores nessa área buscando resolver problemas que quanto mais perto de aplicações reais mais difíceis se tornam.

Como proposta para um trabalho futuro seria interessante a implementação de Algoritmos Genéticos para otimização da Logística dos Correios na Zona da Mata Mineira utilizando o Problema do Caixeiro Viajante através da lógica do Sistema GIDEON em alguma linguagem de programação.

## 11 REFERÊNCIAS BIBLIOGRÁFICAS

ARAKAKI, Reinaldo Gen Ichiro. O Problema de Roteamento de Veículos e Algumas Metaheurísticas. 1998. Monografia apresentada para o Exame de Qualificação do Curso de Computação Aplicada – CAP, Instituto Nacional de Pesquisas Espaciais - INPE. Set.1998. Disponível em: <<http://www.lac.inpe.br/~lorena/reinaldo/monografia.pdf>>. Acesso em 28 jun.2004.

BERTINI, Luciano. O Problema do Caixeiro Viajante. 2003. Segundo Trabalho Prático de P.A.A. – Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, jun.2003. Disponível em: <<http://www.dcc.ufmg.br/bertini/paa/tp2>>. Acesso em 28 jun.2004.

BITTENCOURT, Guilherme. Inteligência Computacional. 2003. Monografia (Graduação em Sistemas de Informação) – Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, Florianópolis, 2003. Disponível em: <<http://www.das.ufsc.br/gia/softcomp/>>. Acessado em jul.2004.

BOAVENTURA NETTO, Paulo Oswaldo. Grafos: Teoria, Modelos, Algoritmos. 2.ed. São Paulo:Edgard Blücher LTDA, 2001.

CITI. História da Inteligência Artificial. Disponível em: <[http://www.citi.pt/educacao\\_final/trab\\_final\\_inteligencia\\_artificial/historia\\_da\\_ia.html](http://www.citi.pt/educacao_final/trab_final_inteligencia_artificial/historia_da_ia.html)>. Acessado em abr.2004.

COSTA, Mateus Conrad B. da. Problema do Caixeiro Viajante Assimétrico.2002 Segundo Trabalho de P.A.A. – Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, set.2003. Disponível em: <<http://www.dcc.ufmg.br/~nivio/cursos/pa02/tp2/tp21/tp21>>. Acesso em 28 jun.2004.

CUNHA, Cláudio Barbieri da. et al. Experimentos Computacionais com Heurísticas de Melhorias para o Problema do Caixeiro Viajante. 2002. Trabalho apresentado no XVI congresso da Anpet – Associação Nacional de Pesquisa e ensino em Transportes. Departamento de Engenharia de Transportes, Escola Politécnica da Universidade de São Paulo, Natal - RN, out.2002. Disponível em: <[http://www.ptr.usp.br/docentes/cbcunha/files/2-opt\\_TSP\\_Anpet\\_2002\\_CBC.pdf](http://www.ptr.usp.br/docentes/cbcunha/files/2-opt_TSP_Anpet_2002_CBC.pdf)>. Acessado em out.2004.

FALQUETO. Em que consiste a Inteligência Artificial. Laboratório de Conexão e Ciências Cognitivas, Universidade Federal de Santa Catarina, Florianópolis, 2004. Disponível em: <[http://www.inf.ufsc.br/~falqueto/aGraduacao/INE5633Sist\\_Intel/IA\\_Geral/IA\\_Introd\\_Historia.PDF](http://www.inf.ufsc.br/~falqueto/aGraduacao/INE5633Sist_Intel/IA_Geral/IA_Introd_Historia.PDF)>. Acessado em abr.2004.

LAUDON, Kenneth C.;LAUDON, Jane Price. Sistemas de Informação com Internet. 4.ed. Rio de Janeiro:LTC, 1999.

MARIA, Rosa. PC Burro. Estado de Minas, Belo Horizonte, 2.set.2004. Caderno de Informática, p.3.

MIRANDA, Márcio Nunes. Algoritmos Genéticos: Fundamentos e Aplicações. 1999. Grupo de Teleinformática e Automação, Universidade Federal do Rio de Janeiro. Disponível em: <<http://www.gta.ufrj.br/~marcio/genetic.html>>. Acessado em nov.2004.

OLIVEIRA, Rafael Bruno C. de. O Problema do Caixeiro Viajante (PCV). 2004. Monografia (Graduação em Ciência da Computação) – Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, jun.2004. Disponível em: <<http://www.dcc.ufmg.br/~rafaelb/trabalhos/aeds3/tp1/pcv/node12.html>>. Acessado em abr.2004.

PALAZZO, Luiz A. M. Algoritmos para Computação Evolutiva. 1997 Relatório Técnico – Grupo de Pesquisa em Inteligência Artificial – Universidade Católica de Pelotas, Pelotas.

POZO, Aurora et al. Computação Evolutiva. 2000. Grupo de Pesquisas em Computação Evolutiva - Departamento de Informática, Universidade Federal do Paraná, Curitiba, 2000. Disponível em: <<http://www.inf.ufpr.br/~aurora/tutoriais/Ceapostila.pdf>>. Acessado em out.2004.

RICH, Elaine; KNIGHT, Kevin. Inteligência Artificial. 2.ed. São Paulo: Makron Books do Brasil LTDA, 1994.

SANTOS, Nilson Moutinho dos. Refletindo Sobre a Inteligência Artificial. 2003. Disponível em: <<http://www.din.uem.br/~nmsantos>>. Acessado em jul.2004

STAIR, Ralph M.; REYNOLDS, George W. Princípios de Sistemas de Informação. 4.ed. Rio de Janeiro: LTC, 2002.

YEPES, Igor. Algoritmos Genéticos (AG's). 2002 Trabalho Individual – Departamento de Ciência da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre,2002. Disponível em: <<http://www.geocities.com/igoryepes/visualizar2.htm>>. Acessado em ago.2004.

## GLOSSÁRIO

**Adjacente:** vizinhança, proximidade, contigüidade.

**Aleatório:** dependente de fatores incertos, sujeitos ao acaso; casual, fortuito, acidental.

**Algoritmos Evolucionários:** usam modelos computacionais de processos evolucionários como elementos chave no projeto e implementação de sistemas para a solução de problemas.

**Algoritmos Genéticos:** são algoritmos matemáticos inspirados nos mecanismos de evolução natural e recombinação genética.

**Ângulos Sementes:** são ângulos que geram descendentes ou ramificações.

**Aptidão:** Habilidade ou capacidade resultante de conhecimentos adquiridos.

**Aresta:** ângulo exterior formado por dois planos que se cortam; esquina, quina, canto.

**Bits:** unidade mínima de informação em um sistema digital, que pode assumir apenas um de dois valores 0 ou 1.

**Caixeiro:** empregado em casa de comércio que vende ao balcão; balconista ou aquele que entrega em domicílio as mercadorias compradas; entregador.

**Codificado:** reunião de leis em código ou aplicação de código.

**Cognitiva:** aquisição de um conhecimento.

**Complexidade:** que abrange ou encerra muitos elementos ou partes.

**Computação Evolucionária:** área de pesquisa interdisciplinar que compreende diversos paradigmas inspirados no princípio Darwiniano da evolução das espécies.

**Cromossomo:** unidade morfológica e fisiológica, visível ou não ao microscópio óptico, e que contém a informação genética.

**Crossover:** recombinação de genes

**Darwinismo:** sistema de história natural cuja conclusão extrema é o parentesco fisiológico e a origem comum de todos os seres vivos, com a formação de novas espécies por um processo de seleção natural.

**Fenótipo:** que tem a mesma aparência.

**Fitness:** Valor de Aptidão

**Fuzzy Logic:** lógica nebulosa

**Gene:** unidade hereditária ou genética, situada no cromossomo, e que determina as características de um indivíduo.

**Genótipo:** composição gamética total do indivíduo ou zigoto.

**Grafo:** diagrama composto de pontos, alguns dos quais são ligados entre si por linhas para representar graficamente conjuntos de elementos inter-relacionados.

**Hardware Evolucionário:** é uma extensão do modelo genético de aprendizado no espaço de estruturas complexas como circuitos eletrônicos.

**Heurística:** conjunto de regras e métodos que conduzem à descoberta, à invenção e à resolução de problemas.

**Híbrido:** originário do cruzamento de espécies diferentes.

**Inferência:** tirar por conclusão; deduzir pelo raciocínio.

**Inteligência Artificial:** é uma área da computação que trabalha com o processamento simbólico de conhecimento, criando programas que fazem os computadores parecerem inteligentes.

**Inteligência Computacional:** área da ciência que estuda a teoria e a aplicação de técnicas inspiradas na Natureza.

**Interação:** ação que se exerce mutuamente entre duas ou mais coisas, ou duas ou mais pessoas; ação recíproca.

**Linear:** que apresenta a disposição de linha; semelhante a esta ou a um traço.

**Lógica Nebulosa:** tem por objetivo modelar o modo aproximado de raciocínio humano, visando desenvolver sistemas computacionais capazes de tomar decisões racionais em um ambiente de incerteza e imprecisão.

**Programação Evolucionária:** é uma estratégia estocástica de otimização, similar a Algoritmos Genéticos, que dispensa a representação genômica de soluções e envolve outros tipos de operadores de mutação.

**Programação Genética:** é uma técnica automática de programação que propicia a evolução de programas de computadores que resolvem (ou aproximadamente resolvem) problemas.

**Redes Neurais:** são modelos computacionais não lineares, inspirados na estrutura e operação do cérebro humano, que procuram reproduzir características humanas.

**Sistemas Especialistas:** são programas computacionais destinados a solucionar problemas em um campo especializado do conhecimento humano.

## ANEXO A

### TERMINOLOGIA BIOLÓGICA

Será traçado um breve paralelo entre a terminologia biológica e a terminologia da Computação Evolutiva dos principais termos empregados neste trabalho, visando auxiliar no entendimento dos conceitos aplicados:

<b>Terminologia Biológica</b>	<b>Terminologia da Computação Evolutiva</b>
<b>ADAPTABILIDADE:</b> Toda característica de um organismo vivo que aumenta as possibilidades de sobrevivência e de deixar descendência no meio em que habita. Também proporcional ao tempo esperado para extinção. “Os adaptados são aqueles que se ajustaram ao Ambiente existente ao seu redor e cujos descendentes se ajustarão a ambientes futuros”. (J. Thoday, - UM Século de Darwin, 1959).	Qualquer alteração na estrutura de um cromossomo (string de caracteres de certo alfabeto) que melhore sua capacidade de resolver determinado problema, permitindo-lhe maiores chances de sobrevivência e reprodução. ·
<b>AJUSTE:</b> Medida que indica o grau de aptidão de um indivíduo ao meio ambiente.	Valor de Adaptabilidade ou Fitness. Valor relativo a avaliação da Função Objetivo de uma solução. Também chamada de função de Avaliação ou Função Objetivo, em problemas de otimização. ·
<b>ALELOS:</b> Se diz que dois ou mais GENES são ALELOS (entre si) ou ALELOMORFOS (um em relação ao outro), quando: 1. Ocupam a mesma posição relativa (LOCUS) em CROMOSSOMOS HOMÓLOGOS e quando, na mesma célula se unem em pares durante a meiose; 2. Produzem efeitos diferentes no mesmo grupo de processos do desenvolvimento; 3. Podem mutar de um para o outro.	Possíveis valores dos genes (no caso de representação binária 0 ou 1). ·
<b>CROMOSSOMO:</b> Corpo em forma de filamento, constituído principalmente por DNA e proteína, que se encontra em número variável no núcleo de todas as células animais e vegetais. Componente genético responsável pelo fenótipo do indivíduo.	Um vetor ou string devidamente codificado, pertencente ao conjunto de possíveis soluções de um determinado problema. ·
<b>DIPLOIDE:</b> Diz-se do núcleo cujos cromossomos se apresentam aos pares, sendo homólogos (contém os mesmos GENES na mesma seqüência) os membros de cada par, de forma que há um número o dobro do número haplóide.	Em cada local do cromossomo existe um par de genes. Isto permite uma boa economia de memória. ·
<b>ESPÉCIE:</b> Grupo de indivíduos inter-cruzantes que são isolados reprodutivamente de outros grupos semelhantes, contendo fenótipos semelhantes.	Indivíduos componentes de uma mesma população.

FENÓTIPO: Conjunto das características de um indivíduo observadas ou discerníveis por outros métodos, fornecidas pelo genótipo.	Expressão física das estruturas. Conjunto decodificado de parâmetros. ·
GENES: Unidade de material hereditário. Unidade básica do cromossomo que define, de acordo com seu valor e posição, uma característica; um determinador hereditário que especifica uma função biológica; uma unidade de herança (DNA) localizada em um lugar fixo no cromossomo.	Variáveis, componentes da string que representa o cromossomo. ·
GENÓTIPO: Arranjo dos genes e cromossomos de um organismo correspondente a determinado fenótipo.	Estruturas atuais. Cromossomos codificados.
HAPLOIDE: Célula que contém um único CROMOSSOMO ou conjunto de CROMOSSOMOS em vez de dois em cada núcleo, cada qual consistindo de uma única seqüência de GENES. Um exemplo é um GAMETA.	Em lugar de um cromossomo existe um AG Modelo Ilha de um gene simples. ·
INDIVÍDUO: Um exemplar de uma espécie que interage com o meio ambiente.	Mesmo que Cromossomo.
MUTAÇÃO: Variação devida a alguma alteração da constituição hereditária com aparecimento de uma nova variedade em qualquer espécie viva.	Modificação aleatória do cromossomo visando gerar novas possíveis soluções à explorar. ·
POPULAÇÃO: Conjunto de indivíduos da mesma espécie; grupos inteiros de organismos de um tipo; um grupo inter-cruzantes de animais ou vegetais; grupo amplo do qual pode-se tomar uma amostra.	Conjunto de possíveis soluções.
REPRODUÇÃO: Multiplicação de novos indivíduos a partir de outros de mesma espécie.	Cruzamento (crossover) vem a ser uma troca de segmentos dos cromossomos (strings) selecionados.
SELEÇÃO NATURAL: Mecanismo que garante aos indivíduos mais aptos, maiores chances de reprodução; fertilidade diferencial na natureza que favorece indivíduos que se adaptam melhor ao meio ambiente e tende a eliminar aqueles não adaptados.	Determinada pela aptidão (fitness) do Indivíduo. Quanto maior a sua aptidão, maiores as chances de gerar descendentes. Indivíduos com aptidões baixas representam soluções indesejáveis no momento da avaliação.

Fonte: YEPES, 2002.

## ANEXO B

### INTELIGÊNCIA ARTIFICIAL OU INTELIGÊNCIA COMPUTACIONAL

Fazendo-se uma busca no “Searcher Altavista” (buscador brasileiro), com o termo “*Inteligência Artificial*”, foi encontrado 3187 sites que tratavam do assunto. Nos sites em inglês os vocábulos “Artificial Intelligence” eram tratados em 128488 sites.

Outra pesquisa, com o termo “*Inteligência Computacional*”, revelou 172 sites sobre o assunto enquanto que os vocábulos “Computational Intelligence” eram tratados em 7120 sites em inglês.

Afinal de contas devemos ainda falar em IA, qual a diferença entre IA e IC?

Um dos sites pesquisados para descobrir essa possível diferença foi o do Núcleo de Pesquisa em Inteligência Computacional Aplicada - DEE - PUC-Rio.

O Departamento de Engenharia Elétrica da PUC-Rio, desde março de 1993, oferece cursos de Inteligência Computacional ao nível de graduação e pós-graduação.

Os cursos introduzem as técnicas computacionais inteligentes em *Redes Neurais, Computação Evolucionária, Lógica Fuzzy e Sistemas Especialistas* em uma abordagem prática, envolvendo a teoria e a aplicação em *sistemas inteligentes* de:

\* previsão; suporte à decisão; controle; otimização; modelagem; classificação e reconhecimento de padrões em geral.

O diagrama da página seguinte apresenta uma visão geral da área de *Computação Natural* que incorpora, entre outras, a linha de pesquisa de Inteligência Computacional.

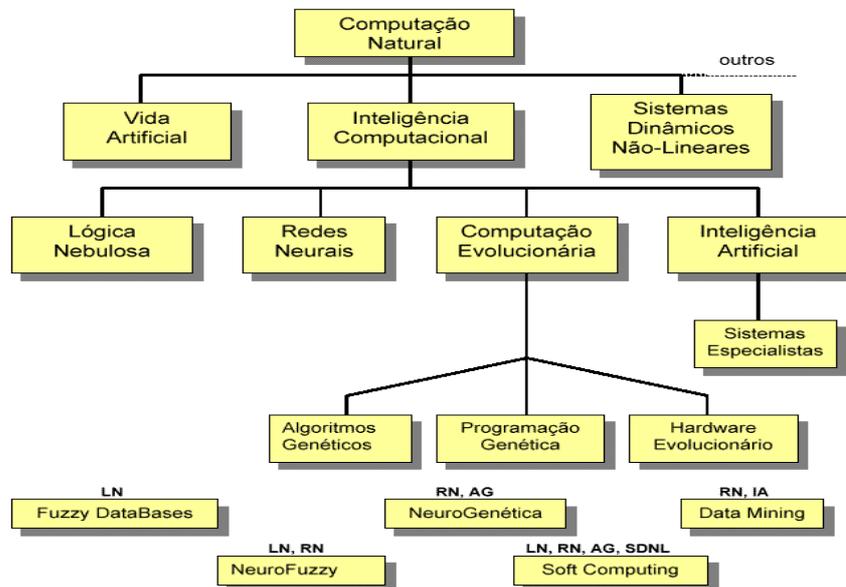
O ICA estabelece o objetivo da Inteligência Computacional como sendo o seguinte:

**“A Inteligência Computacional busca, através de técnicas inspiradas na Natureza, o desenvolvimento de sistemas inteligentes que imitem aspectos do comportamento humano, tais como: aprendizado, percepção, raciocínio, evolução e adaptação”.**

Essa tabela mostra a técnica computacional utilizada e de onde essa técnica foi buscar inspiração na natureza.

<i>Técnica Computacional</i>	<i>Inspiração na Natureza</i>
Redes Neurais	Neurônios biológicos
Computação Evolucionária	Evolução biológica
Lógica Fuzzy	Processamento lingüístico
Sistemas Especialistas	Processo de Inferência

O ICA propõe um organograma fazendo uma taxionomia de Computação Natural, onde Inteligência Artificial aparece como um subcampo de Inteligência computacional (veja abaixo).



Será mesmo a Inteligência Artificial (IA) subcampo da Inteligência Computacional (IC)? Os modelos em que se baseia toda a IA não são bem mais antigos e complexos do que a data do surgimento da IC?

Bem, com toda essas divergências decorrentes da evolução da capacidade humana, será muito difícil para os pesquisadores da área, chegarem a uma resposta única.

## ANEXO C

### O QUE A INTELIGÊNCIA ARTIFICIAL PODE E NÃO PODE FAZER

Um outro aspecto bastante curioso da IA diz respeito ao que realmente pode ser obtido de prático na direção dos seus objetivos.

Conforme Coello, “é provável que ainda estejamos muito longe de alcançar os propósitos finais da IA, que são entender o que é inteligência e construir entidades (máquinas) inteligentes”.

A IA é considerada por muitos uma disciplina em crise pela ausência de um paradigma unificador, que servia de base a pesquisadores e praticantes que desejam construir sistemas inteligentes.

Isso ocorre, em parte, a cargo do que o termo inteligência tende a colocar como referencial para medir o sucesso da área a sua manifestação mais desafiadora, a inteligência humana, ainda em larga medida incompreendida.

Mas esta não é a opinião de todos os pesquisadores da área, existem aqueles que são extremamente otimistas com o futuro da Inteligência Computacional.

A revista *Internetbr* publicou uma entrevista do pesquisador Flávio Joaquim de Souza, PHD em Inteligência Computacional pela PUC-Rio e professor do Departamento de Sistema e Computação da Faculdade de Engenharia da UERJ.

O referido profissional é um entusiasta da integração homem-máquina no seu sentido mais íntimo: preconiza a inserção de um chip de silício de 0,5 por 5 mm, com internet e telefonia implantados na nuca, o que o capacitará fazer conexão mental com outros seres humanos podendo conversar “*telepaticamente*”, enviando e recebendo inclusive imagens, sensações, cheiro e gosto.

Quando perguntado sobre como seria essa telepatia, ele respondeu:

**O cérebro estará conectado ao chip (computador do futuro) via terminações nervosas (neurônios) e, através de pensamentos, mandaremos um comando cerebral por um neurônio que ativará o chip. Este, por sua vez, ligado por telefonia celular à Internet, vai conectar um endereço de um outro chip acoplado à nuca de outra pessoa. Automaticamente estas pessoas vão entrar em contato direto. Passaremos a usar somente o nosso hardware: o cérebro.**

E o entrevistado segue relatando as imensas alterações pelas quais o mundo passará pela aproximação do homem da máquina. Interessante é que ele prevê o desaparecimento de duas profissões:

**Os médicos já estão com os dias contados. Em talvez duas décadas, já teremos sistemas de computadores ligados à internet e a bancos de dados que serão capazes de diagnosticar com muito mais precisão e complexidade que qualquer médico que conhecemos. Hoje, alguns deles já usam sistemas baseados em técnicas de inteligência computacional para diagnosticar. No futuro, o médico vai se transformar em terapeuta e o diagnóstico vai ficar a cargo dos computadores. Para alívio e felicidade de muita gente, outra carreira desnecessária no futuro é a dos políticos. A tendência é de que todo cidadão que quiser auxiliar ou propor uma lei possa colocar o texto na Internet em determinado site e, através de um plebiscito previamente marcado, essa lei, uma vez aprovada, seja incorporada ao sistema. Chama-se democracia participativa.**

Ao ser perguntado sobre a convivência entre o homem e o computador, o entrevistado falou da construção de robôs semelhantes ao homem (andróides) que falarão diretamente com conosco, ouvirão a sua voz e interpretarão seus sentimentos, fazendo coisas que determinarmos.

Então o entrevistador perguntou se esses andróides terão emoções, ao que ele respondeu:

**A gente não pode dizer que a máquina terá emoção. Mas podemos dizer que haverá mecanismos que simulem estados emocionais na máquina. Só assim a máquina pode ser sensível ao estado emocional do homem e ficar cada vez mais próxima e semelhante ao seu criador. Sim (isto é perigoso), isso me preocupa. Tem um artigo da Universidade de Varsóvia que propõe um modelo baseado em lógica nebulosa, uma das técnicas da inteligência computacional, que permitirá ao computador Ter um comportamento variado, mudando conforme seu estado emocional. Desta forma, não poderemos prever todas as situações. Isto nos leva a temer reações por parte da máquina que nos causem danos físicos.**

Seria essa, uma posição realista com relação às possibilidades da IC? Ou é apenas fantasia na cabeça de escritores de ficção científica?

Para verificar isso, foi feita uma rápida pesquisa na Internet à procura de outro testemunho de um pesquisador na área de IC e encontramos Hans Moravec, um Cientista Pesquisador Chefe no Instituto de Robótica na Universidade de Carnegie Mellon, nos EUA, que se interessa por robôs desde criança, tem pontos de vistas intrigantes e pessoais sobre eles e, em sua entrevista concedida à revista online NOVA, em outubro de 1997, ele fala sobre o que pensa sobre a evolução dos robôs:

**Bem, imagine quatro estágios. Penso que estamos apenas na margem de sermos capazes de ver máquinas que trabalham suficientemente bem que se tornarão os predecessores para a primeira geração de robôs produzidos em massa – os**

quais não são brinquedos. Tenho um particular em mente: uma pequena máquina que poderia ser um aspirador de pó o qual, com milhares de MIPS de capacidade computacional, seria capaz de manter um mapa ou imagem tridimensional muito densa de suas redondezas. Seria capaz tanto de planejar suas ações quanto de navegar, de tal maneira que ele soubesse a cada momento onde ele está e é mesmo capaz de identificar as principais peças de mobiliário e itens importantes ao seu redor. Assim – uma pequena máquina, pequena o suficiente para se meter debaixo das coisas e encontrar sua própria estação de recarga de esvaziar o pó acumulado de tempos em tempos. Esta é a pesquisa que estamos fazendo e penso às vezes que em cinco ou dez anos teremos alguma coisa parecida com isso – e seus sucessores se tornarão um pouco mais capazes. Eles terão um pouco mais de dispositivos e serão programados para uma faixa mais ampla de trabalhos até que, eventualmente, você consiga a primeira geração de robôs universais, os quais terão mobilidade e capacidade de entender e manipular o que estiver acontecendo ao seu redor.

No final da entrevista, depois da descrição das quatro gerações cada vez mais evoluídas de robôs, NOVA pergunta: Você pode prever um robô entendendo a psicologia de um terrorista melhor do que um ser humano? E Moravec responde:

Bem, como um desenvolvimento avançado. Estamos falando agora de 40 ou 50 anos a partir de agora, nós já temos a quarta geração dessas máquinas e seus sucessores, os quais eu penso finalmente serão melhores do que os seres humanos, em todos os sentidos possíveis. Mas as duas habilidades que são provavelmente as mais difíceis dos robôs alcançarem, porque elas são as coisas que fazemos melhor, que têm sido questão de vida ou morte para nós, são, primeiro, interagir com o mundo físico. Você sabe, nós temos tido de encontrar nossa comida e evitar nossos predadores e lidar com coisas na base de momento a momento. Assim manipulação, percepção, mobilidade esta é a primeira área. A outra área é a interação social. Porque temos vivido em tribos desde sempre e temos sido hábeis para julgar o intento e o provável comportamento de outros membros de nossa tribo para sermos bem sucedidos. Assim a espécie de intuição social que temos é muito poderosa e provavelmente seu uso está próximo do pleno poder de processamento de nossos cérebros – o equivalente a centenas de trilhões de cálculos por segundo – mais uma porção de conhecimento muito especial, alguns dos quais são inerentes (hard wired) outros são aprendidos à medida que crescemos. Penso que isso é aonde os robôs chegarão por último. Mas, uma vez lá, então eles continuarão evoluindo. Penso que haverá um tempo quando os robôs nos entenderão melhor do que nós nos entendemos a nós mesmos, ou entendemos uns aos outros. E você pode mesmo imaginar o tempo no futuro distante quando os robôs serão capazes de hospedar uma simulação muito detalhada do que está ocorrendo em nosso cérebro e então serem capazes de nos manipular.

## APÊNDICE A

### O ENSINO DE COMPUTAÇÃO QUE OFUSCA AS CIÊNCIAS BÁSICAS

“O Estado de São Paulo de 03 de setembro de 2003 publicou uma reportagem sobre a mudança de comportamento dos estudantes e orientadores acadêmicos americanos em relação às escolhas de cursos nas universidades (SANTOS, 2003).

Cursos tradicionais como química e física estão sendo abandonados em favor de disciplinas do curso de ciência da computação.

Os educadores até chegam a rezear que as Universidades venham a produzir levas de pessoas capazes de usar computadores como ferramentas de pesquisa e cálculos, mas bem poucos capazes de usar o método científico para resolver problemas.

*Gerald Wheeler*, diretor-executivo da Associação Nacional de Professores de Ciências americano, chegou mesmo a afirmar que:

**"Cursos de ciência da computação ensinam prática e técnicas, não ensinam o pensamento crítico como a física o faz".**

*Michael Hammer*, consultor americano bem-sucedido na área de administração, mas que tem título de doutorado em ciência da computação, pensa que:

**"A ciência da computação fornece base intelectual para lidar com coisas de enorme complexidade".**

Sabe-se que a ciência da computação desenvolve a capacidade de pensar em sistemas, ensinando a importância da precisão e oferecendo ainda habilidade para tratar com barganhas e com enormes questões de design.

Assim, vemos que se até mesmo a ciência da computação ainda é um campo de conhecimento que está em acomodação no próprio mundo universitário, quem dirá a área de Inteligência Artificial.

Este fato pode ser observado também pela grande dificuldade que tem a Sociedade Brasileira de Computação, em estabelecer um currículo de referência que possa servir de paradigma para orientar a explosão dos cursos de informática, ciência da computação e engenharia da computação que têm proliferado no Brasil.

As mudanças na área de computação estão acontecendo tão rapidamente que seria temerário e imprudente fazer previsões neste momento sobre um campo tão recente como o da Inteligência Artificial.

Fonte: SANTOS, 2003.

## APÊNDICE B

### QUAIS DISCIPLINAS DEVEM SER INCLUÍDAS NO CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO, DE MANEIRA QUE PROPORCIONE AO ALUNO UMA BOA BASE PARA TRABALHAR NESTA ÁREA, NÃO SÓ COMO PROFISSIONAL, MAS TAMBÉM PARA SEGUIR UM CURSO DE PÓS-GRADUAÇÃO?

A tabela abaixo foi montada a partir de informações contidas no site da PUC do Rio de Janeiro.

Graduação	Pós-graduação
<i>Inteligência Artificial e Data Mining:</i> ELE 1343 - 3 créditos	ELE 2396 - 3 créditos
<i>Redes Neurais:</i> ELE 1346 - 4 créditos - Laboratório SHF 2 h.	ELE 2760 - 3 créditos
<i>Computação Evolucionária:</i> ELE 1347 - 4 créditos - Laboratório SHF 2 h.	ELE 2761 - 3 créditos
<i>Lógica Nebulosa:</i> ELE 1348 - 4 créditos - Laboratório SHF 2 h.	ELE 2762 - 3 créditos
<i>Inteligência Computacional: Aplicações em Finanças, Comércio e Indústria</i> ELE 1349 - 4 créditos - Laboratório SHF 2 h.	
	<i>Sistemas Inteligentes Aplicados:</i> ELE 2763 - 3 créditos

Considerando que nas disciplinas de Inteligência Artificial e Data Mining também é visto “Noções de Prolog e LISP”.

Vamos considerar como exemplo a situação atual da UEM.

Considerando que o curso de Ciência da Computação é seriado, constatamos a existência de uma disciplina, denominada 1018 – Inteligência Artificial, que propõe ministrá-la no período de um ano, mas com os seguintes conteúdos:

\* *Noções Gerais de Inteligência Artificial*: onde é abordado, ao nível de informação, os diversos paradigmas de desenvolvimento de sistemas inteligentes, excetuando o de Agentes Inteligentes;

\* *Linguagens Aplicativas*: onde em nível introdutório são tratados os conceitos básicos, a sintaxe e a programação das linguagens Prolog e Lisp;

\* *Estudo de Representação do Reconhecimento*: abordagem das principais técnicas de representação do conhecimento;

\* *Técnicas de Busca*: onde aprendem os principais métodos de busca, com implementação em Prolog;

\* *Sistemas Especialistas*: onde vêm conceitos básicos, estrutura e aplicações;

\* *Noções de Base de Conhecimento e Tendências da IA*.

Para melhorar esta situação, foi criado um projeto de ensino permanente denominado Grupo de Sistemas Inteligentes (GSI), onde o aluno tem a oportunidade de trabalhar os conceitos básicos daquelas disciplinas que não são abordadas na grade curricular normal do Curso de Ciência de Computação. Contudo, isto já é uma iniciativa particular do professor da disciplina. Informações podem ser obtidas através do site <http://www.din.uem.br/ia>.

Dessa maneira, para que a UEM possa ter uma visão mais abrangente do que é chamado na PUC-Rio de Inteligência Computacional, deve pelo menos contemplar no currículo do curso as disciplinas a título Graduação da PUC-Rio.

Talvez o currículo do curso de ciência da computação deva ser alterado para contemplar a evolução que tem ocorrido na área de Inteligência Artificial, ou da Inteligência Computacional.

## APÊNDICE C

### **SOFTWARE DE PLANEJAMENTO DA LOGÍSTICA DE ENTREGA E AGENDAMENTO REDUZ CUSTOS DE DISTRIBUIÇÃO**

A Lighstone Group possui sede em Mineola, Nova Iorque, e escritórios em Palo Alto, Califórnia; Glastonbury, Connecticut; e Westminster, Maryland – Estados Unidos. A companhia dedica-se ao desenvolvimento de software para logística e para suporte à tomada de decisões empresariais relacionadas à cadeia de suprimentos, especialmente agendamento para seu pessoal de campo e veículo de entrega.

A empresa foi fundada em 1989 por três pessoas que, inicialmente, desenvolveram o sistema de agendamento para a companhia aérea MachUp, posteriormente vendida à American AirLines SABRE Group. Hoje, é uma das líderes em desenvolvimento de software de planejamento de logística e agendamento, tendo Resource in Motion Management System (RiMMS) como seu produto chefe. Uma ampla faixa de indústrias e clientes, grandes e pequenos, desfrutam da economia de custos e do serviço de atendimento aprimorado proporcionados pelo RiMMS.

Por exemplo, a Tuscan-Leigh Dairies agendava suas entregas de modo tradicional, ou seja, a partir de um Atlas, mapas e alfinetes de marcação, até a companhia automatizar seu tedioso processo de organização. Com o software de planejamento da logística de entrega de agendamento da pequena Lighstone Group Inc, a Tuscan espera melhorar o serviço de atendimento a seus 8.000 clientes, distribuídos em seis estados norte-americanos. Muitos clientes requerem que as entregas sejam feitas em um curto espaço de tempo. O software de planejamento da logística de entrega e de agendamento permite que a Tuscan-Leigh Dairies atenda as necessidades de tempo de seus clientes, tarefas especialmente importante para uma empresa cujos produtos são perecíveis. O software elimina rotas sobrepostas, reduzindo quilômetros e tempo de entrega. Embora o sistema custe mais de 30.000 dólares, ele se pagará dentro de um ano.

O sistema inclui dados da Navigation Technologies, fornecedora de bancos de dados de mapas de navegação, e permite que os usuários adicionem seus próprios dados, como rotas,

quantidade de entregas e exigências dos clientes. O software, então, determina quem deve efetuar a entrega e qual a melhor rota. Calcula, também, o melhor tempo de entrega e outros detalhes referentes a cada parada no trajeto. Este sistema está preparado para lidar, inclusive, com eventuais mudanças nos meios de captação e de entrega.

O uso do RiMMS melhora a eficiência do planejamento das rotas, otimiza a quantidade de paradas por dia e aumenta a satisfação do cliente. O sistema é projetado para empresas que requeiram prestar serviços de alto nível a seus clientes, como serviço de entrega, serviço de campo, captação e gerenciamento da equipe de vendas. O sistema pode contabilizar vários tipos de veículos, diversos depósitos, diferentes localidades de início e fim para os veículos, janelas de tempo, múltipla capacidade de obstáculos, níveis de habilidade dos motoristas, entre outras variáveis.

Em abril de 1998, a Lighstone anunciou a Easy Router, o primeiro serviço de planejamento de logística baseado na internet. Utilizando a engrenagem de planejamento do RiMMS para gerar rotas otimizadas, o Easy Router possibilita que seus assinantes recebam, com rapidez, rotas otimizadas e agendamentos com localização de mapas e direções a seguir pela internet. Os assinantes enviam a informação referente a uma entrega específica, via internet, à medida que necessário, e o Easy Router retorna seu serviço em uma hora.

Fonte: STAIR, 2002.

## **APÊNDICE D**

### **INTELIGÊNCIA ARTIFICIAL: UMA FORMA INTELIGENTE DE AGENDAR TAREFAS NA VOLVO**

A adaptação evolutiva só recentemente foi aplicada ao mundo dos computadores e do software. Uma vez escrito o programa, ele age ou executa da mesma forma cada vez que executado sob as mesmas condições.

Um desenvolvimento excitante no campo da inteligência artificial está modificando modo como os programas operam. O campo dessas mudanças é chamado de algoritmos genéticos. Nesse ramo da inteligência artificial, é estudado o mecanismo como um programa responde e como suas saídas evoluem ou são modificadas com o passar do tempo.

Tendo se tornado um projeto de pesquisa da ciência da computação, os algoritmos genéticos estão sendo aplicados nas corporações, incluindo a fabricante de carros e caminhões sueca Volvo.

A Volvo, conhecida por promover a segurança de seus carros e caminhões, tem sido pioneira na indústria automotiva por décadas. Foi uma das primeiras empresas a fazer experiências com equipes de trabalho, quando os outros fabricantes de automóveis ainda viam seus trabalhadores como engrenagens substituíveis de uma linha de montagem.

Assim, não foi surpresa quando a inovadora fabricante de carros e caminhões lançou mão dos algoritmos genéticos como forma de ajudá-la a agendar tarefas específicas de uma de suas unidades de produção.

Embora conhecida primeiramente por seus carros de luxo, a Volvo também é uma bem sucedida fabricante de caminhões. Os motoristas profissionais de caminhão transportando cargas de um lugar para outro, usualmente gastam a maior parte de seus dias e noites na frente do volante.

Por anos, os compradores de carros exigiam opções de escolha na seleção das características interiores e exteriores. O mesmo é verdade para os motoristas de caminhão. Gastando muito tempo na boléia, esses caminhoneiros queriam interiores personalizados, para

tornar as longas horas na direção de um caminhão mais confortáveis e agradáveis. Colocar essa flexibilidade num caminhão é uma alegria para os caminhoneiros, mas uma dor de cabeça para os fabricantes. A questão está em programar as tarefas e as instalações de produção.

A fábrica de caminhões da Volvo em Dublin, Virginia, possui mais de 90.000 metros quadrados de instalações de produção para fabricar cabines de trailers comerciais. As restrições sobre o que pode ser produzido e em quais quantidades são excessivamente complexas. Adicionar flexibilidade para produzir cabines personalizadas torna o problema ainda pior. Como fazer para agendar tarefas e instalações de produção para assegurar flexibilidade ao processo produtivo? Para a Volvo, a resposta foi o uso de algoritmos genéticos.

Para resolver seus problemas de agendamento, a Volvo decidiu testar uma nova tecnologia. Comprou um programa, chamado OPTIFLEX, que usa a teoria de algoritmos genéticos. O programa permite que a programação “evolua”, continuamente melhorando uma série de agendamentos aceitáveis. Ele combina e varia as agendas e seleciona a melhor. É como a teoria da evolução de Darwin, que confia na variação e na seleção natural.

A principal diferença é que o computador varia agendamentos e seleciona o melhor. O resultado é uma boa agenda, que evolui das programações passadas. O processo é automatizado; de acordo com um gerente, o usuário diz ao programa o que quer produzir e, então, vai tomar um café. Quando volta, o OPTIFLEX apresenta uma agenda para a produção.

Fonte: STAIR, 2002.

## APÊNDICE E

### DARWINISMO ELETRÔNICO

Qualquer um que se preocupe com o fato de o mundo estar se movendo depressa demais rumo a ser consumido com o desenvolvimento da tecnologia, enquanto deixa de lado o processo natural de vida, pode se confortar com uma recente tendência de resolução de problemas por computador. Empresas de software se uniram a cientistas com o propósito de criar programas utilizando processos de resolução de problemas do mundo natural como modelo.

Os algoritmos genéticos entraram no software de computadores porque os cientistas e desenvolvedores perceberam que as soluções para um certo número de suas dificuldades já existiam na natureza, especialmente relacionadas à genética e a seleção natural. A Thinking Tools Inc.; de Monterey, Califórnia, está trabalhando com a Texas Instruments para construir um sistema que ajudará as empresas de transportes na distribuição de carga para áreas distantes. Seu modelo para projeto não é outro senão o salmão comum, que encontra a melhor rota para seu destino, um local de desova. As duas empresas planejam equipar as embalagens com pequenas telas que indicariam seu destino e itinerário para os trabalhadores. As unidades também poderiam receber informações do computador central da empresa de transportes, de forma que poderiam alterar a rota no caso de um bloqueio na estrada.

O FacePrints, criado pelo Professor Victor S. Johnson, do departamento de psicologia da New México State University, utiliza algoritmos genéticos para auxiliar testemunhas a identificar suspeitos criminais. As testemunhas muitas vezes não conseguem descrever as características individuais de um suspeito, porém se saem muito melhor no reconhecimento de rostos. O FacePrints funciona por meio de ilustrações aleatórias de faces em uma tela de computador, combinando e recombinando características até surgir a melhor descrição.

O FacePrints consiste em centenas de características individuais, como um nariz de pugilista ou sobrancelhas cerradas, cada uma das quais representada como uma seqüência digital de código de computador. O sistema capacita a testemunha de um crime a selecionar a figura mais exata de um suspeito, um passo mais perto da seleção natural, a partir de um

grupo de 30 imagens composto por características escolhidas aleatoriamente. O FacePrints então combina a melhor figura com outras 30 faces construídas aleatoriamente e repete o processo seguidamente enquanto elimina as características improváveis. Em um determinado momento, uma foto muito útil do suspeito é formada. “Uma face em particular possui um bilhão de bilhões de possibilidades”, observa Johnson, o inventor do FacePrints. O FacePrints pode pesquisar um enorme “espaço de faces” muito rapidamente.

Enquanto o ser humano pode se cansar depois de realizar uma dúzia de tentativas para resolver um problema, um algoritmo genético continua incansavelmente, tentando milhões de soluções até encontrar uma resposta. A maioria será descartada instantaneamente. Milhares, no entanto poderão ser abordagens promissoras sobre as quais nenhum ser humano pensou antes. Por exemplo, quando os projetistas da General Electric foram solicitados a planejar uma pá de hélice mais eficiente para os motores dos jatos 777 da Boeing Co., eles se depararam com um desorientante número de possibilidades. Os fatores que afetavam o desempenho e o custo de motores a jato chegavam a um número com 129 trios de zeros! Isso exigia um supercomputador executando bilhões de cálculos por segundo e bilhões de anos para testar todas as combinações. No entanto, algoritmos genéticos suplementando um sistema especialista resolveram o problema em menos de uma semana.

O sistema híbrido, denominado Engeneous, começa com um grupo de “cromossomos” digitais, cada um representando um fator de projeto. Esses cromossomos se combinam a fim de criar dezenas de modelos hipotéticos. Os melhores modelos recebem permissão para se “desenvolver”, para trocar “genes” e dar origem a uma nova geração de soluções, algumas das quais ainda melhores. As mais apropriadas são selecionadas para procriar novamente, e assim por diante até que bons projetos sejam produzidos. Em apenas três dias, o Engeneous criou um modelo que aumentou a eficiência do motor de 1%, um aperfeiçoamento significativo no campo dos motores a jato.

Em outra aplicação, o Engeneous aumentou 5% a eficiência de uma nova hélice para turbinas de usinas de força. Surpreendentemente, o algoritmo genético violou algumas instruções de projeto que foram utilizadas pelo sistema especialista. Enquanto os humanos tendem a acreditar que o que funcionou antes é o melhor caminho a seguir, os algoritmos genéticos não possuem tais pressuposições, de modo que eles podem apresentar soluções que os humanos rejeitariam sem sequer verificá-las. Depois que as pessoas entendem como o algoritmo fez isso, elas podem absorver o conhecimento para novas instruções de projetos.

Em princípio, os algoritmos genéticos poderiam ser capazes de encontrar respostas para problemas demasiadamente complexos para serem definidos por pessoas. Sistemas “inteligentes” comuns são baseados em alguma idéia preconcebida de como proceder, algum tipo de modelo do mundo real. Porém esse modelo pode ser enganoso, posto que as pessoas não compreendem perfeitamente as forças que movem a maior parte dos eventos mais complexos do mundo real. Os algoritmos genéticos, como as redes neurais, procuram soluções e padrões na modalidade de baixo para cima. Esses métodos de “modelo livre” podem conduzir a novas avenidas do conhecimento.

Fonte: LAUDON, 1999.