

SÉRGIO MAURÍCIO TRAD JÚNIOR

**MONOGRAFIA DE CONCLUSÃO DE CURSO:
PROBLEMAS DE OTIMIZAÇÃO**

Trabalho de conclusão de curso apresentado ao Curso de Ciência da Computação.

UNIVERSIDADE PRESIDENTE ANTÔNIO CARLOS

Orientador: Prof. Eduardo Macedo Bhering

BARBACENA

2003

SÉRGIO MAURÍCIO TRAD JÚNIOR

**MONOGRAFIA DE CONCLUSÃO DE CURSO:
PROBLEMAS DE OTIMIZAÇÃO**

Este trabalho de conclusão de curso foi julgado adequado à obtenção do grau de Bacharel em Ciência da Computação e aprovado em sua forma final pelo Curso de Ciência da Computação da Universidade Presidente Antônio Carlos.

Barbacena – MG, 03 de junho de 2003.

Eduardo Macedo Bhering - Orientador do Trabalho

Prof^ª. Lorena Sophia C. de Oliveira - Membro da Banca Examinadora

Prof^ª. Alessandra Marques de Carvalho - Membro da Banca Examinadora

AGRADECIMENTOS

Agradeço ao meu avô Antônio, aos meus pais e familiares, amigos da faculdade aos mestres que me iluminaram no caminho da saber, ao professor Eduardo que me auxiliou com grande esmero na produção de todo trabalho, e em todos aqueles que acreditaram no meu potencial.

RESUMO

Este trabalho apresenta um algoritmo de otimização para o planejamento de redes de distribuição considerando o objetivo de minimização dos custos de implantação e o objetivo de maximização da confiabilidade de operação. O método proposto é um algoritmo genético adaptado para a abordagem multicritério, visando a determinação do conjunto de soluções Pareto-ótimas. Foram usados operadores de cruzamento e mutação adequados para lidar com o problema, bem como uma função de aptidão penalizada, chegando assim em uma solução viável próxima da ótima.

Palavras-chave: Otimização, algoritmos genéticos, redes de distribuição.

SUMÁRIO

<u>SUMÁRIO.....</u>	<u>6</u>
<u>LISTAS.....</u>	<u>7</u>
<u>INTRODUÇÃO.....</u>	<u>9</u>
<u>1 PESQUISA E REVISÃO BIBLIOGRÁFICA.....</u>	<u>10</u>
<u>2 A HEURÍSTICA ALGORITMO GENÉTICO COMO ALTERNATIVA PARA O PROBLEMA DE OTIMIZAÇÃO DE REDES DE DISTRIBUIÇÃO COM CRITÉRIOS MÚLTIPLOS.....</u>	<u>27</u>
<u>4 METODOLOGIA UTILIZADA.....</u>	<u>35</u>
<u>5 CONCLUSÕES.....</u>	<u>41</u>
<u>6 REFERÊNCIAS BIBLIOGRÁFICAS.....</u>	<u>43</u>

LISTAS

FIGURA 1 - CLASSIFICAÇÃO DAS ABORDAGENS APROXIMATIVAS.....	12
FIGURA 2 - EXEMPLO DA ESTRUTURA DE UM CROMOSSOMO.....	16
FIGURA 3 - ESQUEMA DE CODIFICAÇÃO BINÁRIA DE UM CROMOSSOMO.	16
FIGURA 4 - ESQUEMA DA CODIFICAÇÃO REAL E ALFABÉTICA DE UM CROMOSSOMO.....	17
FIGURA 5 - ESQUEMA GENÉRICO PARA REPRESENTAR AS HABITANTES E SEUS RESPECTIVOS PONTOS DE DISTRIBUIÇÃO.....	30
FIGURA 6 - REPRESENTAÇÃO DOS HABITANTES E SEUS RESPECTIVOS PONTOS DE DISTRIBUIÇÃO PARA O PROBLEMA PROPOSTO.	30
FIGURA 7 - ESQUEMA DE UM CROMOSSOMO COM CODIFICAÇÃO BINÁRIA PARA O PROBLEMA PROPOSTO.....	31
FIGURA 8 - ESQUEMA DE GERAÇÃO DE CROMOSSOMOS VIÁVEIS SEGUNDO A RESTRIÇÃO DE SINGULARIDADE.....	31
FIGURA 9 - CROMOSSOMO COM SEUS RESPECTIVOS LÓCUS E POSIÇÃO DAS INTERFACES ENTRES UNIDADES CONSUMIDORAS DISTINTAS.....	32
FIGURA 10 - GERAÇÃO ALEATORIAMENTE DE UMA POPULAÇÃO INICIAL COM CINCO INDIVÍDUOS.....	33
FIGURA 11 – RELAÇÃO DE ALUNOS PARA TESTE DE DISTRIBUIÇÃO.....	36

FIGURA 12 – RELAÇÃO DE ESCOLAS COM SUA CAPACIDADE MÁXIMA.....	36
FIGURA 13 – RELAÇÃO DAS DISTÂNCIAS DOS ALUNOS AS ESCOLAS.....	37
FIGURA 14 - VALORES DA FITNESS OBTIDOS PELO MÉTODO DE SELEÇÃO PROPORCIONAL.	
38	
FIGURA 15 - VALORES DA FITNESS OBTIDOS PELO MÉTODO DE SELEÇÃO TOURNAMENT..	39
FIGURA 16 – MELHOR CONFIGURAÇÃO OBTIDA (FITNESS 131,22) NOS TESTES.....	40

INTRODUÇÃO

Muitos problemas interessantes de otimização combinatória não podem ser resolvidos numa quantidade de tempo razoável pelos algoritmos exatos tradicionais como: “branch-and-bound”, “implicit enumeration” e “cutting plane” (Ignizio e Cavaller , 1994 , citados por Nobre, 1999).

Conseqüentemente, para a resolução destes problemas, pode-se utilizar heurísticas (IGNÁCIO et al.,2000). Nos últimos anos tem-se verificado o surgimento de um significativo conjunto de técnicas e algoritmos computacionais muito eficientes, mas que não garantem a solução ótima do problema da programação inteira. Esses algoritmos são denominados de heurísticos ou aproximativos (GOLDBARG e LUNA, 2000). As heurísticas mais promissoras para resolver problemas combinatoriais em diversas áreas incluem: Simulated Annealing (KIRKPATRICK, 1983), Algoritmos Genéticos (HOLLAND, 1975) e Busca Tabu (GLOVER e LAGUNA, 1997). Até então, existem poucos trabalhos reportando a utilização de Algoritmos Genéticos com aplicações em problemas de distribuição geográfica a partir de pontos de pré-definidos. Considerando as inúmeras aplicações com sucesso dos algoritmos genéticos em outras áreas, acredita-se que estes possam igualmente ser aplicados em problemas de quadro de horários, distribuição e logística de grandes empresas, etc.

1 PESQUISA E REVISÃO BIBLIOGRÁFICA

1.1. O QUE É HEURÍSTICA?

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

Segundo Zanakis et al.(1981), citado por IGNÁCIO et al.(2000) uma heurística pode ser definida como “um procedimento simples e rápido, geralmente baseada no sentido

comum e, se supõe, que ofereça uma boa solução (não necessariamente a ótima) no tratamento de problemas complexos”.

De acordo com Ignízio e Cavalier (1994), citados por Rodriguez (1996), as heurísticas são as mais promissoras abordagens para solução de problemas de programação inteira. Lembram ainda que, enquanto os algoritmos exatos para solução de problemas de programação inteira garantem uma solução ótima para certos tipos de problemas, os métodos heurísticos não podem provar a otimalidade das suas soluções, mas oferecem, entretanto, soluções aceitáveis, inclusive para problemas complexos e de grande porte, com baixo custo computacional.

Segundo GOLDBARG e LUNA (2000), Os primeiros métodos gerais clássicos estavam associados a estratégias de enumeração incompleta, solução parcial ou relaxações e dentre muitas classificações possíveis para as heurísticas, sugeriu a constante da Figura 1.

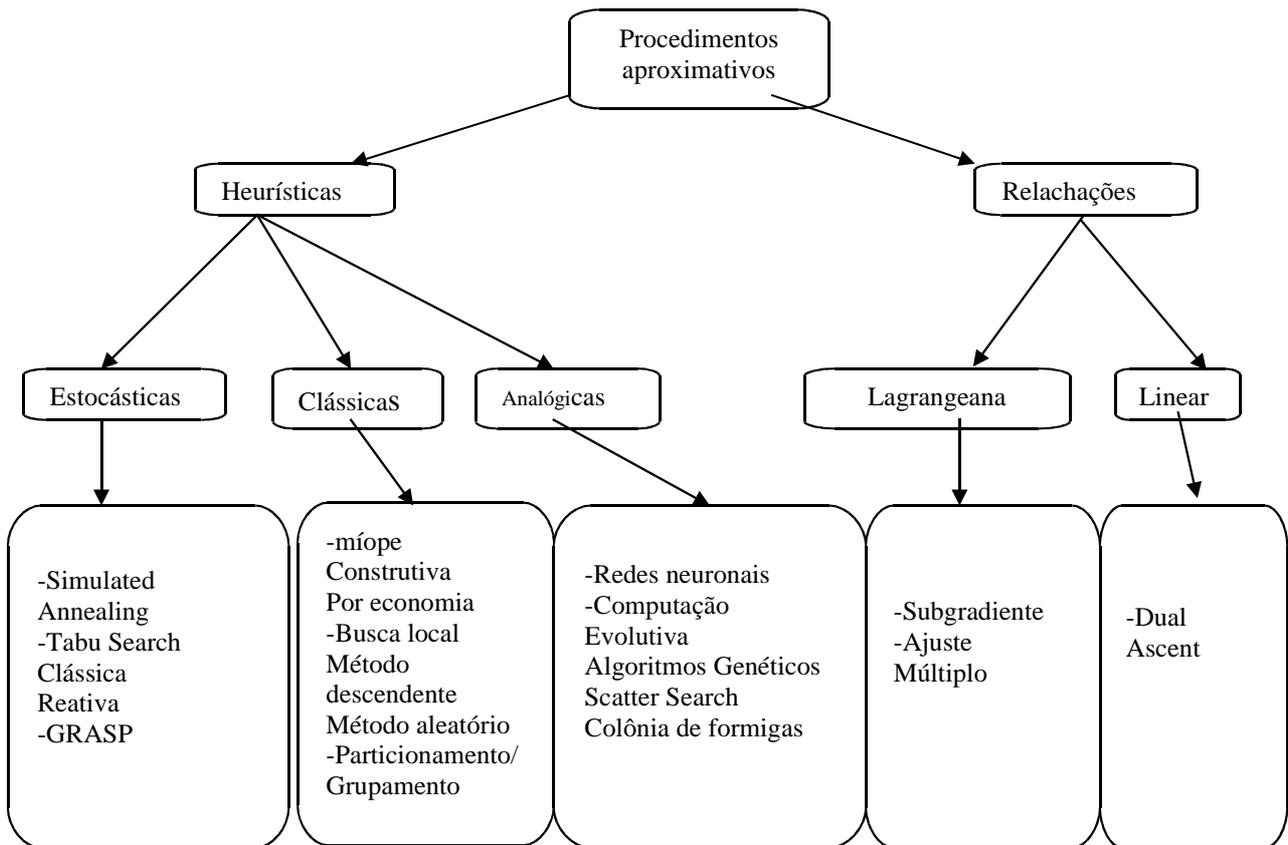


Figura 1 - Classificação das abordagens aproximativas

A Figura 1 resume as várias abordagens conhecidas para os procedimentos heurísticos de solução. As heurísticas da linha clássica exploram, de modo casual, a estrutura do problema sem definir claramente uma estratégia universal de solução (GOLDBARG e LUNA, 2000).

As heurísticas iniciaram historicamente o seu desenvolvimento a partir de problemas específicos. As primeiras heurísticas relatadas na literatura eram destinadas à solução de problemas específicos e não eram, via de regra, passíveis de serem utilizadas em outros problemas; as heurísticas clássicas de roteamento são um bom exemplo disso (GOLDBARG e LUNA, 2000). Entretanto, nos últimos 30 anos, pesquisadores da área de Pesquisa Operacional adaptaram idéias de outras áreas no desenvolvimento de metaheurísticas (SOUZA e moccelin, 2000). Segundo GOLDBARG e LUNA (2000), o termo metaheurística é reservado para um grupo especial das chamadas heurísticas estocásticas e analógicas. As metaheurísticas são técnicas que, superpondo-se a métodos heurísticos de

busca de vizinhança guiam a busca com vistas à superação da otimalidade local e à obtenção de soluções de qualidade superior (SOUZA E MOCCELLIN, 2000). Dentre as mais populares, inclui-se Simulated Annealing (Kirkpartrick, 1983), Algoritmos Genéticos (HOLLAND, 1975) e Busca Tabu (GLOVER et. al., 1997). Segundo SOUZA e MOCCELLIN (2000), tais métodos consistem de procedimentos de busca no espaço de soluções, definidos por estratégias que exploram apropriadamente a topologia de tal espaço. Segundo este mesmo autor, o sucesso das metaheurísticas se deve a fatores como: i) alusão a mecanismos de otimização da natureza (nos casos do Algoritmo Genético e Simulated Annealing); ii) aplicabilidade geral da abordagem, i.e. existe invariavelmente uma estratégia geral de solução, cabendo apenas adaptá-la ao caso específico; iii) facilidade de implementação; e iv) qualidade da solução obtida aliada a um esforço computacional relativamente baixo.

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

1.2. ALGORITMOS GENÉTICOS

Os algoritmos genéticos são procedimentos de busca que têm sido usados na solução dos mais variados problemas de diferentes domínios, como programação automática, ecologia, pesquisa operacional e etc (GARCIA et al., 2000). Para REBELLO e HAMACHER (2000), os AG's são métodos robustos, que podem ser utilizados para resolver problemas em pesquisa numérica, otimização de funções e aprendizagem de máquina, entre outras aplicações. Eles têm sido apontados como uma técnica promissora para resolver problemas combinatoriais, como os clássicos problemas de Programação Inteira. Apesar de não garantirem otimalidade, sua grande vantagem está no melhor desempenho ou performance computacional, além de ser um procedimento de relativa simplicidade. Segundo GARCIA et

al. (2000), praticamente todos os problemas NP-completos ou NP-difíceis têm uma versão de solução heurística que usa esse tipo de algoritmo.

De acordo com(GARCIA et al. 2000), os Ag`s foram desenvolvidos por John Holland e seu grupo de pesquisas, mas a teoria sobre algoritmos genéticos só foi formalizada em 1975 com o aparecimento do livro *Adaptation in Natural and artificial systems* de John Holland. Posteriormente GOLDBARG (1989) aperfeiçoou a idéias desenvolvidas por Holland.

Os Ag`s são métodos de busca e otimização baseados nos princípios de seleção natural e reprodução genética. Eles empregam um processo adaptativo e paralelo de busca de soluções em problemas complexos e se enquadram na classe de métodos heurísticos inteligentes ou Metaheurísticas (REBELLO e HAMACHER, 2000).

1.2.1. PRINCÍPIOS BÁSICOS

O algoritmo genético inicia o processo de otimização a partir de um conjunto de configurações (população inicial) que pode ser obtida aleatoriamente ou usando algoritmos heurísticos construtivos simples e rápidos (ROMERO e GALLEGO, 2000).

Em cada iteração é obtido um novo conjunto de configurações (nova população) a partir da população corrente usando os operadores de seleção, crossover e mutação. Em cada nova iteração são encontradas configurações de melhor qualidade e, eventualmente, nesse processo iterativo pode ser encontrada a solução (configuração) ótima global. Segundo ROMERO e GALLEGO (2000), um algoritmo genético realiza uma busca usando um conjunto de soluções (configurações) e através de um processo iterativo são encontradas novas configurações candidatas. Ainda, de acordo com ROMERO e GALLEGO (2000) o número de configurações visitadas nesse processo de busca deve ser um número muito reduzido de configurações do espaço de configurações e deve existir uma estratégia adequada para visitar as configurações mais atrativas.

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

Algoritmo Genético
Início
 Gerar uma população inicial
 Avaliar a fitness dos indivíduos da população
 Repetir
 Início
 Selecionar um conjunto de pais na população
 Cruzar os pais de modo que se reproduzam
 Avaliar a fitness dos filhos gerados
 Substituir os filhos julgados inadequados
 Fim
 Até quando critério de parada seja atendido
Fim

2.1.2. ELEMENTOS DO ALGORITMO GENÉTICO

Independente da sofisticação de um AG, cinco componentes que são básicos em qualquer implementação de AG (IGNÁCIO et al. (2000):

1. Representação das soluções do problema em termos de cromossomos.
2. Modo de criar a configuração da população inicial.
3. Uma função de avaliação (aptidão) que permita ordenar os cromossomos, de acordo com a função objetivo.
4. Operadores genéticos que permitam alterar a composição dos novos cromossomos gerados pelos pais, durante a reprodução.
5. Valores dos parâmetros que o AG usa (tamanho da população; probabilidades associadas com a aplicação dos operadores genéticos: critério de seleção, critério de sobrevivência dos cromossomos; taxa de mutação; critérios de parada; etc.)

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

2.2.2.1. Codificação das soluções do problema em termos de cromossomos

Na terminologia dos AG's um cromossomo representa um indivíduo na população (i.e. uma configuração ou solução), definido normalmente como um vetor de componentes (GOLDBARG e LUNA, 2000). Cada cromossomo apresenta os seguintes elementos: genes, alelo, locus e fitness (Figura .2). Os genes (G) representam as componentes do cromossomo (i.e. uma variável do problema). Os alelos descrevem os possíveis estados de um atributo do indivíduo (i.e. os possíveis valores de uma variável do problema). Em problemas com variáveis binárias tem-se $G \in \{0,1\}$. Finalmente, os locus representam a N-ésima posição do atributo no cromossomo (i.e. a posição da componente no vetor dos componentes). A Fitness é a medida da aptidão do indivíduo (cromossomo). Normalmente associada ao valor da função objetivo para uma dada solução.

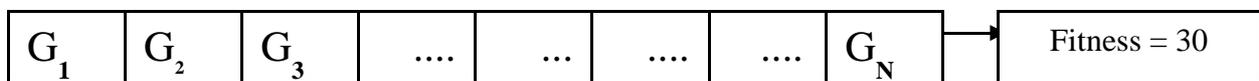


Figura 2 - Exemplo da estrutura de um cromossomo

A representação desses cromossomos é apontada pelos especialistas como um fator determinante no sucesso do AG. A forma de codificação mais difundida é a codificação binária. Nesta codificação, cada cromossomo é representado por uma seqüência inteira de bits (Figura 3).



Figura 3 - Esquema de codificação binária de um cromossomo.

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

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

Entretanto, em muitas situações a codificação binária é inadequada para representar a solução do problema, como na representação de pesos em redes neurais, seqüências de proteínas, rotas de veículos etc. Portanto, outras formas de codificação devem ser utilizadas, para tais propósitos, como a codificação alfabética e a codificação real. Na codificação alfabética, os alelos são representados por uma seqüência de caracteres alfabéticos, enquanto na codificação real, tais alelos são representados por seqüências de números reais (Figura 4).

Codificação alfabética

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

Codificação real

4	8	0,5	7	1	3	4	8
---	---	-----	---	---	---	---	---

Figura 4 - Esquema da codificação real e alfabética de um cromossomo.

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

1.2.2.2 População inicial

O algoritmo genético inicia o processo de otimização a partir de um conjunto de configurações (população inicial). A população em AG refere-se a um conjunto de indivíduos ou um conjunto de soluções do problema (GOLDBARG e LUNA, 2000). Em relação à população existem dois aspectos que devem ser especificados. O tamanho da população N e a forma em que é determinada a população inicial. O tamanho da população deve ser escolhida levando em conta o tamanho do problema e os outros parâmetros genéticos, como a taxa de Crossover e a taxa de mutação (ROMERO e GALLEGO, 2000).

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

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

1.2.2.3 Seleção

Após decidir sobre a codificação a ser adotada e a geração da população inicial ter sido efetuada para o algoritmo genético, a decisão seguinte refere-se a forma de executar a

seleção, isto é, como escolher os indivíduos “pais” na população que gerará os novos descendentes ou “filhos” para próxima geração.

Uma função de aptidão (fitness) é utilizada para quantificar a qualidade genética dos Cromossomos-pais, e geralmente corresponde à função de custo em problemas de otimização combinatória (IGNACIO, et al, 2000). De acordo com IGNACIO et al. (2000), esta função de aptidão também é utilizada para decidir se um cromossomo gerado, através de um crossover, substitui ou não um cromossomo reprodutor. O propósito de escolher cromossomos-pais para o cruzamento é incrementar a probabilidade de reproduzir elementos da população que tenha bons valores na função objetivo. Um indivíduo com um valor de aptidão alto possui uma maior probabilidade de contribuir com um ou mais filhos na próxima geração.

Segundo MITCHELL (1996), seleção muito forte reduz a diversidade da população necessária para sua evolução, ocorrendo a chamada “convergência prematura” do algoritmo, por outro lado, seleção muito fraca resultará em evolução muito lenta da população, o que significa uma convergência lenta do algoritmo.

De acordo com GARCIA et al (2000), no processo de otimização quando se implementa o algoritmo genético, a seleção é a etapa mais crítica, pois é a que consome mais tempo com cálculos da função de aptidão e probabilidade de sobrevivência de cada indivíduo. Desta forma, quanto maior o número de indivíduos, maior tempo será gasto no processo de seleção, e conseqüentemente, a velocidade de convergência do algoritmo genético está relacionada diretamente como tamanho da população (GARCIA et al.,2000). Ainda, segundo o mesmo autor, grande parte do custo computacional do algoritmo está concentrada na fase de seleção.

Diversos esquemas de seleção tem sido propostos na literatura de AG. Isso é ainda uma questão aberta para AG's, sendo que a experiência e os ensaios são ainda a melhor forma para selecionar um dado método (MITCHELL,1996).

1.2.2.4 Método de seleção

Segundo De Jong (1975), citado por MITCHELL (1996), o método de seleção proporcional é um dos muitos métodos de seleção que força o AG a reter alguns dos melhores indivíduos em cada geração. Tais indivíduos poderiam ser perdidos se eles forem submetidos a crossover ou mutação. Muitos pesquisadores argumentam que método de seleção pode melhorar significativamente a performance do algoritmo genético (MITCHELL, 1996).

1.2.2.5 Seleção proporcional

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

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

Em que f_i é a função objetivo avaliada no indivíduo i .

Uma vez que f_i deve ser sempre positiva, se a função objetivo for negativa para algum problema específico, o que pode ocorrer normalmente, se constrói um operador $\Psi : f \rightarrow f'$ seja sempre positiva. A função f' é denominada função de aptidão.

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

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

Ao se utilizar a seleção proporcional, deve-se evitar a existência de “super indivíduos” no início do processo, proporcionado por cromossoma que possuem probabilidades de sobrevivência são muito maiores do que a média da população (MITCHELL, 1996 e GARCIA et al, 2000). A existência desses indivíduos é indesejável, uma vez que eles podem convergir para uma solução logo no início do processo. Essa situação recebe o nome de “convergência prematura”.

GARCIA, et. Al, (2000) propõem uma forma de evitar esse problema utilizando a transformação linear de F dada por:

$F(f) = f' = af + b$, que deve satisfazer as seguintes condições:

- i) O valor médio de f' deve ser igual ao valor médio de f .
- ii) O maior valor de f' deve ser no máximo um múltiplo pré estabelecido de média de f .

1.2.2.6 Seleção por escalonamento

A seleção por escalonamento utiliza a função “sigma scalling” para calcular a expectativa de sobrevivência dos indivíduos.

Segundo MITCHELL (1996), a vantagem desse método é evitar a convergência prematura mantendo a pressão de seleção constante durante toda a busca. Como no início da corrida, o desvio padrão da população é geralmente maior, haverá uma penalização para os indivíduos que apresentarem fitness muito acima da média tendendo a reduzir as diferenças de

probabilidade de sobrevivência entre indivíduos mais aptos e menos aptos. Igualmente, no fim da corrida, onde geralmente reduz-se o desvio padrão da população, os indivíduos mais aptos terão maior chance de sobrevivência, permitindo a evolução continuar.

1.2.2.7 Seleção de Boltzman

A seleção por escalonamento mantém a pressão de seleção constante ao longo da corrida. Entretanto, segundo MITCHELL (1996), freqüentemente a variação na pressão de seleção ao longo da corrida pode ser uma importante estratégia para melhorar a performance do AG.

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

1.2.2.8 Seleção por Ranqueamento

Seleção por ranqueamento é um método cujo propósito é também prevenir convergência muito rápida (MITCHELL, 2000). Segundo esse mesmo autor, na versão proposta por BAKER (1985), os indivíduos na população são ranqueados de acordo com o seu fitness, e a probabilidade de sobrevivência de cada indivíduo depende de seu rank ao invés de seu fitness absoluto. As vantagens deste método, segundo MITCHELL (1996) é que ele diminui a pressão de seleção permitindo que indivíduos menos aptos possam reproduzir, uma vez que os valores absolutos do fitness não são considerados, e sim a sua posição no rank. Como desvantagens, pode haver problemas de convergência do algoritmo, com uma baixa pressão de seleção. Outra desvantagem apontada é grande custo computacional necessário para classificar (ranquear) os indivíduos em cada geração.

1.2.2.9 Seleção Tournament

Neste método, também denominado por seleção por jogos, dois indivíduos são escolhidos ao acaso na população. Escolhe-se um parâmetro K . em seguida, um número aleatório r é então escolhido entre 0 e 1. se $r < k$, os dois indivíduos são selecionados para serem pais; caso contrário os dois são então recolocados na população inicial e podem ser selecionados novamente.

Segundo MITCHELL, as vantagens deste método, além de sua simplicidade, é a economia no processamento de operações como cálculo de funções fitness, cálculo de médias, desvio padrão, classificação de indivíduos etc, típicos dos outros métodos de seleção. Com isso, um menor esforço computacional é requerido. Entretanto, por requerer poucas informações do sistema pode aprender pouco sobre as suas características, resultando em baixa performance do algoritmo em termos da qualidade da solução obtida.

1.2.2.10 Operadores genéticos

Operadores Genéticos: são as regras que permitem a manipulação dos cromossomos que, basicamente são (GOLDBARG e LUNA, 2000): Cruzamento (crossover) e mutação.

- Crossover

Este operador permite a obtenção de indivíduos filhos a partir da combinação (cruzamento) dos cromossomos dos pais (GOLDBARG e LUNA, 2000). O procedimento é simples, escolhem-se os cromossomos pais, agrupa-se em dois e cruza-os para formar dois cromossomos filhos, que terão grande probabilidade de ter uma melhor imagem quando aplicados a função objetivo. O operador de crossover, juntamente com a mutação, é

responsável pela evolução dos indivíduos. O cruzamento é feito na esperança de que dois pais considerados aptos como soluções transmitem aos seus filhos as suas características boas, gerando filhos melhores do que eles, e com melhores chances da combinação de características ser uma configuração ótima para o problema (REBELLO e HAMACHER, 2000).

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

- Mutaç o

A muta o   o operador que permite a produ o de um novo indiv duo por altera es aleat rias diretas no cromossomo. Segundo IGNACIO et al. (2000), ainda que por muito tempo a muta o tenha sido concebida com um operador secund rio, ela representa um aspecto importante do AG. De acordo com GARCIA et al. (2000), o prop sito da opera o de muta o   manter a diversidade na popula o, evitando que ela convirja muito rapidamente para um m nimo local (converg ncia prematura). Segundo este mesmo autor, ela permite tamb m que o algoritmo gen tico possa gerar ou recuperar informa es que poder o ser valiosas. Ainda, segundo GARCIA et al. (2000), o operador de muta o deve ser usado com cautela, pois uma taxa de muta o alta aumenta a possibilidade de um “bom” indiv duo ser destru do.

A muta o   uma altera o aleat ria do valor de uma posi o do cromossomo, ou seja, o valor de um determinado gene do cromossomo   invertido. No caso bin rio, a muta o consiste em substituir, com probabilidade P_m (taxa de muta o), o valor de um bit. Para outros tipos de codifica o geralmente   poss vel definir outras alternativas de muta o (ROMERO e GALLEGO, 2000). A muta o n o passa por testes de aptid o uma vez que o objetivo   introduzir diversidade na popula o (IGNACIO, 2000).

1.2.2.11 Parâmetros de controle de um algoritmo genético

A eficiência de um algoritmo genético é altamente dependente de seus parâmetros de controles, tais como o tamanho da população, taxa de crossover e a taxa de mutação. Existem muitos trabalhos desenvolvidos com a finalidade de dimensionar os valores dos parâmetros na literatura de computação evolucionária (MITCHELL, 1996). Entretanto, não há resultados conclusivos sobre os melhores valores desses parâmetros para todas as aplicações. Muitas aplicações usam valores reportados em outros trabalhos, outras definem os seus próprios valores com base em experimentação. Segundo GARCIA et al. (2000), há consenso na literatura de AG que em geral o tamanho de cada população não deve ser menor que 25, a probabilidade de cruzamento deve estar compreendida entre 0,5 e 0,95 e probabilidade de mutação ser menor que 0,01. Segundo MITCHELL (1996), as abordagens que parecem ser muitas promissoras é ter os valores dos parâmetros adotados em tempo real continuamente na busca. Segundo o mesmo autor, tem havido diversas abordagens adaptativas dos parâmetros de AG que têm sido um grande foco de pesquisas pelas comunidades de computação evolucionária.

1.2.2.12 Formalização do Algoritmo Genético

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

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

Onde P é uma população de N indivíduos, $P = (S_1, S_2, \dots, S_n)$. Cada indivíduo S_i , $i = 1..N$ é um conjunto de valores representando uma solução para o problema. F é a função Fitness ou Adequação que retorna um valor real e positivo na avaliação do indivíduo. Θ é o operador de seleção que permite escolher r indivíduos de P. Ω é o conjunto de operadores genéticos que inclui o operador de Crossover - Ω_c e o operador de Mutação - Ω_m . ψ é o

operador de remoção que permite eliminar indivíduos da população para que novos indivíduos mais adaptados sejam adicionados. τ é o critério de parada.

2 A HEURÍSTICA ALGORITMO GENÉTICO COMO ALTERNATIVA PARA O PROBLEMA DE OTIMIZAÇÃO DE REDES DE DISTRIBUIÇÃO COM CRITÉRIOS MÚLTIPLOS

O problema do planejamento ótimo de implantação ou expansão de uma rede de distribuição qualquer envolve o atendimento de vários requisitos (solução multicritério). No planejamento de novas linhas de distribuição de recursos em pontos pré-determinados, ou alteração das existentes, os objetivos de minimização de custos, bem como as regras nas quais as linhas de distribuições destes recursos devem seguir se destacam por possuírem um alto grau de complexidade dependendo dos critérios envolvidos. Desta forma o Algoritmo Genético pode ser uma ferramenta com grande potencial nessa área e necessita ser investigado, principalmente pelo fato de não ter sido levantado nesta pesquisa bibliográfica nenhum trabalho em problemas específicos na área de redes de distribuição por critérios múltiplos.

Abordagens tradicionais constroem uma única função - objetivo como soma ponderada dos diversos objetivos ou requisitos para o problema envolvido, levando a uma solução próxima da ótima. A referência (Carvalho et al. 2001) apresenta um apanhado de diversas abordagens tradicionais, e mostra um algoritmo genético mono-objetivo aplicado à otimização de redes. A estrutura do problema, no entanto, implica em um conjunto de soluções Pareto-ótimas, expressando a natureza do compromisso entre os diferentes objetivos.

A forma mais adequada de tratar esse problema é a rotina computacional fornecer tal conjunto de soluções (Pareto-ótimas), sendo deixada para o projetista a decisão quanto à solução mais adequada, no contexto do problema real.

Os objetivos principais deste trabalho representam minimizar as distâncias de habitantes de uma dada região aos respectivos pontos de distribuições nos quais cada habitante deve ser alocado, observando como critério de distribuição a capacidade máxima de habitantes por ponto (multicritério) .

2.1 FUNÇÃO DE APTIDÃO OU FITNESS

A função de aptidão para o problema proposto pode ser definida através de uma função de aptidão penalizada. Desta forma, os indivíduos que violam alguma restrição do problema terão sua expectativa de sobrevivência E_i reduzida.

FUNÇÃO OBJETIVO

Minimizar

$$\sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij}$$

RESTRIÇÃO DE CAPACIDADE MÁXIMA

$$\sum_{j=1}^N \sum_{i=1}^M X_{ij} \leq P_j$$

RESTRICÃO DE SINGULARIDADE

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

$$X_{ij} = \begin{cases} 1, & \text{se o ponto de distribuição } j \text{ for alocado para o habitante } i. \\ 0, & \text{caso contrário} \end{cases}$$

FUNÇÃO OBJETIVO PENALIZADA

$$F_p = K \sum_{i=1}^M \sum_{j=1}^N C_{ij} X_{ij} - \sum_{j=1}^N CV \times R_j$$

em que:

C_{ij} = é o valor do inverso da distância de cada habitante i , segundo o alternativa j , unidade de ponto de distribuição.

CV = é o valor da constante de penalização.

K = valor da constante de multiplicação da função objetivo. Utilizada para projetar o valor da função objetivo de modo a garantir que o resultado da função de penalização seja sempre positivo.

R_j = valor da violação da restrição de capacidade máxima para o j -ésimo ponto de distribuição. Dado por :

Caso não ocorra violação seu valor é igual a 0 (zero);

Caso ocorra violação seu valor é igual ao número de habitantes que excederam a capacidade máxima P_j do j -ésimo ponto de distribuição.

P_j = é o valor da capacidade máxima de habitantes para o ponto de distribuição j .

M = Número total de Pontos de distribuição.

N = Número total de alternativas de pontos de distribuição para o i -ésimo habitante.

3.1. CODIFICAÇÃO UTILIZADA

Seguindo a formulação apresentada anteriormente, a variável de decisão (VD) X_{ij} representa se o ponto de distribuição “ j ” está alocado ao habitante i . Sendo o número de habitantes “ i ” ($i = 1, 2, \dots, M$), e número de pontos de distribuição j ($j=1, 2, \dots, N$) para cada habitante i , o esquema para representar estes dois atributos é mostrado na figura 5.

(i)	1				2				...				M			
(j)	1	2	...	N	1	2	...	N	1	2	...	N	1	2	...	N

Figura 5 - Esquema genérico para representar as habitantes e seus respectivos pontos de distribuição.

Considerando o problema com $M = 4$ e $N = 3$ (constante para todo M), uma representação desta instância, com as respectivas variáveis de decisão (VD) pode ser esquematizado conforme figura 6.

I	1			2			3			4		
VD	X_{11}	X_{12}	X_{13}	X_{21}	X_{22}	X_{23}	X_{31}	X_{32}	X_{33}	X_{41}	X_{42}	X_{43}

Figura 6 - Representação dos habitantes e seus respectivos pontos de distribuição para o problema proposto.

Conforme já visto, um cromossomo representa uma solução (S_i) viável para o problema. Dentre as formas de codificação de um cromossomo, a codificação binária é a ideal para representar o cromossomo para o problema proposto, uma vez que a natureza de suas

variáveis de decisão também é binária. Nesta representação cada Gene representa um ponto de distribuição alocado a um habitante (X_{ij}).

Aos possíveis estados assumidos por cada gene (variável de decisão), dá-se o nome de alelo. No modelo de programação inteira proposta, os possíveis estados assumidos pelas variáveis são “0” ou “1”, portanto, uma representação de um cromossomo é mostrada abaixo na figura 7.

UC i	1			2			3			4		
Cromossomo	0	1	0	1	0	1	1	1	0	1	0	0

Figura 7 - Esquema de um cromossomo com codificação binária para o problema proposto.

Há uma restrição no modelo de otimização requerendo a alocação de um único ponto de distribuição para cada habitante. Portanto, o esquema da figura 7 é uma solução inviável para o problema, já que os habitantes 2 e 3 possuem duas alocações assinaladas, respectivamente (X_{21} e X_{23}) no habitante 2 e (X_{31} e X_{32}), no habitante 3. há pelo menos duas formas de contornar o problema. Uma delas é gerar aleatoriamente os cromossomos, sem considerar “a priori” a restrição de singularidade. Assim sendo, cada gente (X_{ij}) teria a mesma probabilidade de ser assinalado com os valores 0 e 1, e somente após a geração do cromossomo este seria testado na restrição de singularidade. Esta não parece ser uma alternativa eficiente, pois se cada gene (X_{ij}) tem a mesma probabilidade de ser assinalado com valor 0 ou 1, parece ser bem difícil obter um cromossomo viável segundo a restrição de singularidade. A outra alternativa é adotar um esquema de geração do cromossomo de forma que a restrição de singularidade seja atendida. Utilizando este raciocínio, cada unidade consumidora teria um de seus genes assinalado com o alelo igual a 1 e os demais com alelos iguais a 0. conforme figura 8.

UC i	1			2			3			4		
Cromossomo	0	1	0	1	0	0	0	0	1	0	1	0

Figura 8 - Esquema de geração de cromossomos viáveis segundo a restrição de singularidade.

A partir de agora, apresentaremos apenas o cromossomo, com a sua seqüência de alelos. Os lócus do cromossomo que representam unidades consumidoras distintas serão representadas em cores diferenciadas. A i -ésima posição ($i = 1, 2, \dots, n - 1$) representa a interface entre duas unidades consumidoras sendo um atributo de grande importância em diversas operações do algoritmo genético, conforme será discutido posteriormente. (figura 9)

Posição (i)			1				2			3		4	
0	1	0	1	0	0	0	0	1	0	1	0		

Figura 9 - cromossomo com seus respectivos lócus e posição das interfaces entre unidades consumidoras distintas

3.2. GERAÇÃO DA POPULAÇÃO INICIAL

Após escolhido a codificação apropriada, deve-se gerar um conjunto de indivíduos para iniciar o processo de busca do melhor indivíduo da população. Neste trabalho, os cromossomos serão gerados aleatoriamente. Desta forma, escolhe-se aleatoriamente uma variável de decisão (gene) em cada habitante para receber o valor “1” e assinalando para as demais variáveis daquele habitante, o valor “0”, satisfazendo a restrição de singularidade. Repete-se este processo até que todas unidades consumidoras tenham sido completadas.

O tamanho da população varia, dentre outros fatores, com o tamanho do problema. Há algumas fórmulas indicadas para dimensionar esta população, e serão mencionadas oportunamente. Para o exemplo proposto, poderiam ser gerados os seguintes cromossomos para uma população de 5 indivíduos

0	1	0	1	0	0	0	0	1	0	1	0
0	0	1	0	1	0	1	0	0	0	1	0
1	0	0	1	0	0	1	0	0	1	0	0

0	0	1	0	1	0	0	1	0	0	0	1
0	1	0	0	1	0	1	0	0	0	0	1

Figura 10 - Geração aleatoriamente de uma população inicial com cinco indivíduos

3.3. SELEÇÃO

O objetivo principal da seleção é escolher quais indivíduos da população irão se reproduzir (cromossomos pais). Tais indivíduos são selecionados segundo algum critério. Neste trabalho utilizou-se o método de seleção proporcional onde cada indivíduo i possui uma probabilidade de sobrevivência P_i associada ao seu valor de aptidão (MITCHELL, 1996), dada por :

$$P_i = f_{i_p} \div \sum_{i=1}^N f_{i_p} \times 100$$

em que

f_{i_p} é a função objetivo penalizada avaliada no indivíduo i .

P_i é a probabilidade do i -ésimo indivíduo em percentagem.

O método da roleta russa foi utilizado para implementar a seleção proporcional dos indivíduos. Esse método consiste em associar a cada indivíduo uma fatia da roleta igual a sua probabilidade de sobrevivência P_i . Para implementação computacional deste método procede-se da seguinte forma:

Seja C_i a classe de probabilidade do indivíduo i definida pelo limite inferior (L_i) e limite superior (L_s). Iniciando com L_i igual a zero para o indivíduo i , e assinalando os indivíduos i ($i = 1, 2, \dots, N$) seqüencialmente na roleta, tem-se que o intervalo de probabilidade de um dado indivíduo definido por $L_s = L_i + P_i$ e $L_i = L_s - P_i$.

Tendo sido definidas as classes de probabilidade dos indivíduos, gera-se um número aleatório r ($0 < r < 1$). Uma vez que a probabilidade e as suas respectivas classes para cada indivíduo são expressas em percentagem, toma-se o valor $r \times 100$. Em cada rodada da roleta o indivíduo selecionado será aquele cujo número aleatório gerado estiver contido na sua respectiva classe de probabilidade.

Também foi disponibilizado neste trabalho o método de seleção Tournament. Para implementá-lo foi utilizado o mesmo parâmetro taxa de reprodução, que é expresso em percentual, dividido por cem, afim de obtermos o parâmetro requerido K entre zero e um. Sendo assim a cada dois indivíduos selecionados é gerado um número r aleatório também entre zero e um. Se $r < K$ os indivíduos são selecionados para serem em pais; caso contrário são recolocados na população original e podem ser selecionados novamente.

3.4. CRITÉRIO DE PARADA

Qualquer procedimento iterativo requer a definição de um critério de parada. Os critérios de parada utilizados para este trabalho serão:

- A primeira solução viável encontrada;
- A primeira solução viável encontrada mais n interações;
- Caso não encontre solução viável, executa um máximo de m interações.

4 METODOLOGIA UTILIZADA

Apesar do Algoritmo Genético ser um procedimento de relativa simplicidade, ele requer um número de operações e cálculos bastante elevado, principalmente para problemas de maior porte. Dessa forma sua implementação em computador é fundamental para viabilizar sua aplicação prática. A linguagem de programação que tem sido mais utilizada, segundo as principais fontes de referência sobre o assunto, tem sido a linguagem C. Entretanto, segundo discussões entre especialistas em análise de sistema, esta não parece ser uma condição limitante, e desta forma outras linguagens podem ser utilizadas.

Sendo assim, para implementação deste projeto foi utilizada a linguagem Pascal Orientada a Objetos disponibilizada pelo ambiente de desenvolvimento Delphi 7.0. O critério adotado para utilização deste ambiente foi utilizar uma linguagem que além de robusta e amplamente utilizada pela comunidade acadêmica, disponibiliza-se também mecanismos para um desenvolvimento rápido e eficiente.

4.2. RESULTADOS E DISCUSSÕES

Para ilustrar a aplicação do algoritmo genético foi utilizada uma instância simples de um problema de distribuição de alunos em escolas que consiste em minimizar a distância dos alunos às escolas, sujeito à restrição de capacidade de alunos por escola.

O seguinte cenário foi utilizado para efeito de testes e comparações entre os métodos de seleção implementados.

Índice	Nome
1	ADALMARIO DIAS FERREIRA
2	ALEXANDRE VASCONCELOS COSTA
3	ANDREA DE VASCONCELOS
4	ARLINDA LUIZA OLIVEIRA REIS
5	ARLINDO SILVERIO DE SOUZA
6	AROLDO DE VASCONCELOS
7	AROLDO SILVA
8	DINACIO DE SOUZA LIMA
9	ERCILIA NUNES DE ARAUJO
10	FABIO ROSA LIBERATO

Figura 11 – Relação de alunos para teste de distribuição.

Índice	Nome	Capacidade
1	E. E. Antônio Carlos	0
2	E. E. Dr. Mariano da Rocha	2
3	E. E. Barrinha	5
4	E. E. João Rodrigues	3

Figura 12 – Relação de escolas com sua capacidade máxima.

Índice Aluno	Índice Escola	Distância
1	1	4066,02
1	2	10176,1
1	3	12857,96
1	4	18980,83
2	1	5954,06
2	2	665,84
2	3	3347,7
2	4	9727,33
3	1	7100,03
3	2	11632,46

3	3	14314,32
3	4	20437,2
4	1	5398,62
4	2	711,46
4	3	3393,32
4	4	9646,96
5	1	6978,19
5	2	5693,54
5	3	4073,44
5	4	15889,65
6	1	6011,76
6	2	98,32
6	3	2780,18
6	4	10097,8
7	1	14141,99
7	2	8031,91
7	3	8552,14
7	4	15818,81
8	1	6327,55
8	2	1608,86
8	3	4290,72
8	4	8857,87
9	1	9455,26
9	2	4866,93
9	3	3246,84
9	4	15063,05
10	1	6254,19
10	2	1021,29
10	3	3703,15
10	4	9278,19

Figura 13 – Relação das distâncias dos alunos as escolas.

O algoritmo genético proposto foi testado utilizando uma população inicial de tamanho 20, taxa de reprodução de 95% e taxa de mutação de 20%. Os referidos valores citados foram utilizados tomando por base os valores médios encontrados na literatura. Porém é sabido que tais valores devem ser testados e comparados a fim de obtermos os melhores parâmetros para a execução do algoritmo.

Sendo assim foram comparadas as médias de dez execuções de cada um dos métodos de seleção implementados para o esclarecimento sobre a eficiência e eficácia dos mesmos no contexto do problema.

A figura 14 resume os valores da função de aptidão para o método de seleção proporcional e a figura 15 resume os valores para o método tournament.

Execução	Valor da Fitness	Interações
1	130,94	140
2	123,74	345
3	129,85	175
4	128,14	110
5	126,95	108
6	130,80	101
7	129,97	477
8	128,11	161
9	130,94	105
10	129,35	169
Média	117,16	171,91

Figura 14 - Valores da fitness obtidos pelo método de seleção proporcional.

Execução	Valor da Fitness	Interações
1	26,79	104
2	129,32	150
3	125,20	119
4	32,32	177
5	32,09	129
6	31,46	111
7	131,22	145
8	17,37	113
9	114,86	112
10	30,07	308
Média	60,97	133,45

Figura 15 - Valores da fitness obtidos pelo método de seleção tournament.

Com base nos testes efetuados, podemos concluir que o método de seleção proporcional comparado ao tournament apresenta os melhores resultados para a função de aptidão. Sendo assim tal método pode ser definido como o melhor método de seleção para o problema proposto, uma vez que comparadas às médias dos valores da fitness de cada método de seleção, a proporcional obteve um valor muito superior ao tournament.

No entanto é interessante observarmos que o método de seleção tournament foi capaz de encontrar soluções viáveis com uma média ligeiramente inferior a média do método proporcional.

Índice Aluno	Índice Escola
1	4
2	2
3	4
4	3
5	3
6	2
7	3
8	4
9	3
10	3

Figura 16 – Melhor Configuração obtida (fitness 131,22) nos testes.

5 CONCLUSÕES

Este trabalho teve como objetivo principal desenvolver um sistema de alocação de recursos baseado em critérios com o uso do algoritmo genético. Para ilustrar o procedimento do algoritmo foi proposto um problema de pequeno porte consistindo da minimização das distâncias de alunos as escolas sujeitas ao número de vagas (capacidade) em cada uma delas. Esse modelo de solução para redes de distribuição, pode ser usado em vários outros problemas, como a distribuição de consumidores em supermercados, , pacientes em leitos e hospitais, etc.

O algoritmo genético mostrou ser uma alternativa viável para resolver problemas de redes de distribuição de natureza combinatória, apesar da relativa simplicidade e pequeno número de variáveis do problema proposto.

A performance do algoritmo genético é influenciada, dentre outros fatores, pelos parâmetros : População Inicial, Taxa de Reprodução e Taxa de Mutação. O efeito nas variações destes parâmetros para o problema proposto podem ser testados em uma futura expansão deste trabalho.

O cenário testado apresentou uma boa eficácia e eficiência para os testes realizados, ou seja, alcançou soluções viáveis para o problema. Sendo assim, fica em aberto a necessidade de comparar as soluções obtidas neste trabalho com implementações similares, afim de obtermos um maior detalhamento quanto a qualidade das soluções obtidas. Percebe-se

também, que a eficácia e eficiência da heurística empregada depende, fundamentalmente, de um exaustivo grau de experimentação para que seus parâmetros sejam configurados com a máxima precisão.

6 REFERÊNCIAS BIBLIOGRÁFICAS

[artigo]GARCIA, A.L.; GARCIA, B.; FRIEDMANN, C.V. – **Resolução de um problema de equilíbrio de trabalho e distribuição de agentes de serviço.** In. SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa. Anais. Viçosa: UFV, 2000. p 923-935.

[livro]GOLBARG, M.C.; GOUVÊA, E.F. **Otimização combinatória e programação linear: modelos e algoritmos.** Rio de Janeiro, Campos, 2000. p 649

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

[livro]JOHNSON, K. N., SCHEURMAN, H. L. **Techniques for prescribing optimal timber harvest and investment under different objectives – discussion and synthesis.** Forest Science, Washington, v.18 , 1997.

[livro]MITCHELL, M. **An introduction to genetic algorithms.** London, A Bradford Book, 1996

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

[artigo]ROMERO, R; GALLEGO, R. A. **Algoritmo genético especializado para problemas de alocação ótima de capacitores em sistemas de distribuição de energia elétrica.** In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 32, 2000, Viçosa: UFV, 2000;

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