



UNIVERSIDADE
ESTADUAL DE LONDRINA

Departamento de Computação
Trabalho de Conclusão de Curso

RENAN DUPAS DE MATOS

**UTILIZAÇÃO DE ALGORITMO GENÉTICO PARA
RESOLUÇÃO DO PROBLEMA DE GERAÇÃO DE
HORÁRIOS**

Londrina

2007

RENAN DUPAS DE MATOS

**UTILIZAÇÃO DE ALGORITMO GENÉTICO PARA
RESOLUÇÃO DO PROBLEMA DE GERAÇÃO DE
HORÁRIOS**

Trabalho apresentado ao Curso de
Ciência da Computação, da Universidade
Estadual de Londrina, como requisito à
obtenção do título de Bacharel.

Orientador: Prof. Dr. Pedro Paulo da Silva Ayrosa

Londrina

2007

RENAN DUPAS DE MATOS

**UTILIZAÇÃO DE ALGORITMO GENÉTICO PARA
RESOLUÇÃO DO PROBLEMA DE GERAÇÃO DE
HORÁRIOS**

Trabalho apresentado ao Curso de
Ciência da Computação, da Universidade
Estadual de Londrina, como requisito à
obtenção do título de Bacharel.

COMISSÃO EXAMINADORA

Prof. Dr. Pedro Paulo da Silva Ayrosa.
Universidade Estadual de Londrina

Prof. Dra. Maria Angélica de O. C.
Brunetto
Universidade Estadual de Londrina

Prof. Ms. Rafael Robson Negrão
Universidade Estadual de Londrina

Londrina, 13 de novembro de 2007

A Deus, minha família...

e a minha eterna namorada...

AGRADECIMENTOS

À minha namorada Tatiana que me ajudou e muito para que esse trabalho fosse concluído.

À minha família pelo apoio e suporte.

Aos meus amigos, pela amizade nesses anos de curso.

Ao orientador Prof. Dr. Pedro Paulo, pelo auxílio nesse trabalho.

MATOS, Renan D. **Utilização de algoritmo genético para resolução do problema de geração de horários**. 2007. Trabalho de Conclusão de Curso. Universidade Estadual de Londrina, Londrina.

RESUMO

O Problema de Geração de Horários, conhecido na literatura como *Timetabling Problem*, trata da alocação de horários para todas as aulas de uma determinada escola, universidade etc., considerando-se um conjunto restrito de horários e satisfazendo um conjunto de restrições. O desenvolvimento de uma solução para o problema de forma manual, além de ser trabalhosa e lenta, pode gerar soluções não tão boas. Algoritmos genéticos têm sido utilizados com eficácia para resolução desse problema em tempo aceitável. Neste trabalho, é proposto uma pesquisa e o desenvolvimento de uma solução aceitável, segundo algumas restrições, para o problema de geração de grade horária utilizando a abordagem da computação evolutiva, através do uso de Algoritmos Genéticos.

Palavras-chave: Algoritmos genéticos. Problema de geração de horários. Inteligência artificial.

MATOS, Renan D. **Use of genetic algorithm for solving the problem of generation of schedules.** 2007. Trabalho de Conclusão de Curso. Universidade Estadual de Londrina, Londrina.

ABSTRACT

The problem of generation of schedules, known in the literature as Timetabling Problem, is treated as a problem in that we need to allocate schedules for all classes of a certain school, university etc, having a restrict set of schedules and to please a set of constraints. The development of a solution for the problem of manual way, besides being hard and slow, can generate solutions other than expected. Genetic algorithms have been used with effectiveness for this problem resolution in an acceptable time. In this work, is proposed a research and a development of an acceptable solution, according to some constraints, for timetabling problem using the evolutionary computing approach, through the use of genetic algorithms.

Key-words: Genetic algorithms. Timetabling problem. Artificial intelligence.

LISTA DE ILUSTRAÇÕES

Figura 1: Algoritmo Genético.....	5
Figura 2: Cruzamento em um ponto.....	8
Figura 3: Cruzamento em dois pontos	9
Figura 4: Mutação	9
Figura 5: Indivíduo [1].....	14
Figura 6 Indivíduo [2].....	15
Figura 7: Indivíduo [3].....	15
Figura 8: Indivíduo [4].....	15
Figura 9: Indivíduo A antes do cruzamento	17
Figura 10: Indivíduo B antes do cruzamento.....	17
Figura 11: Posições dos indivíduos A e B para serem cruzadas.....	18
Figura 12: Indivíduo C após a troca das disciplinas iguais.....	18
Figura 13: Indivíduo D após a troca das disciplinas iguais.....	18
Figura 14: Indivíduo C após o cruzamento.....	19
Figura 15: Indivíduo D após o cruzamento.....	19
Figura 16: Indivíduo A para cruzamento em dois pontos	20
Figura 17: Indivíduo B para cruzamento em dois pontos	20
Figura 18: Indivíduo A depois do cruzamento em dois pontos	20
Figura 19: Indivíduo antes da mutação	21
Figura 20: Indivíduo após a mutação	21

SUMÁRIO

1	INTRODUÇÃO	1
1.1	CONSIDERAÇÕES INICIAIS	1
1.2	OBJETIVOS E JUSTIFICATIVAS	2
1.3	ORGANIZAÇÃO DO TRABALHO	2
2	ALGORITMOS GENÉTICOS	4
2.1	DEFINIÇÃO.....	4
2.2	REPRESENTAÇÃO CROMOSSÔMICA	6
2.3	DEFINIÇÃO DA POPULAÇÃO INICIAL	6
2.4	OPERADORES GENÉTICOS.....	7
2.4.1	Seleção.....	7
2.4.2	Cruzamento	8
2.4.3	Mutação	9
2.5	CRITÉRIOS DE PARADA.....	10
3	O PROBLEMA DE GERAÇÃO DE HORÁRIOS	11
3.1	DESCRIÇÃO DO PROBLEMA.....	11
3.2	TRABALHOS CORRELATOS.....	12
3.3	MODELO CONSIDERADO	12
4	ABORDAGEM DO PROBLEMA DE GERAÇÃO DE HORÁRIOS POR ALGORITMOS GENÉTICOS.....	14
4.1	REPRESENTAÇÃO DA SOLUÇÃO.....	14
4.2	GERAÇÃO DA POPULAÇÃO INICIAL.....	16
4.3	SELEÇÃO	16

4.4	CRUZAMENTO	16
4.4.1	Cruzamento em um Ponto	17
4.4.2	Cruzamento em dois Pontos	19
4.5	MUTAÇÃO	21
4.6	CRITÉRIO DE PARADA	22
5	CONCLUSÃO.....	23
	REFERÊNCIAS.....	24

1 INTRODUÇÃO

Este capítulo visa apresentar algumas considerações sobre o problema de construção de grade de horários, assim como abordar os objetivos e justificativas desse trabalho. Também é mencionado a organização do mesmo.

1.1 CONSIDERAÇÕES INICIAIS

Em Instituições de Ensino Superior, todo ano letivo ocorre a montagem de uma grade de horários. Essa programação é uma tarefa importante, mas também difícil, visto o grande número de possibilidades e as necessidades de se respeitar algumas restrições. Dessa forma:

[...] o problema de construção de grades horárias pode ser visto como relações entre conjuntos de professores, disciplinas, alunos, períodos de tempo e recursos. Estas relações estão sujeitas ainda a uma série de restrições, cujas combinações aumentam a complexidade do problema (JUNIOR, 2000, p. 1).

Essas instituições despendem de dias para chegar a um horário definitivo, o qual, ainda muitas vezes ou é inviável, ou não se adequam à realidade. Devido à complexidade desse problema de otimização combinatória, sua resolução por algoritmos exatos em tempo hábil se torna difícil, pois, como argumenta Cicon (2006, p. 1), “a família de problemas ‘Timetabling’ se caracteriza por um espaço de busca muito grande acompanhado de um conjunto de restrições que devem ser respeitadas”.

Dessa forma, a resolução desse problema por técnicas heurísticas, as quais mesmo não tendo a garantia de se encontrar a solução ótima, se mostram mais eficazes, pois localizam uma solução boa em um tempo curto de processamento, pois conseguem encontrar boas soluções. O objetivo desse trabalho é apresentar um Algoritmo Genético, que é uma técnica heurística que utiliza conceitos advindos

das Teorias de Evolução e Genética, para resolução do Problema da construção de Grade Horária.

1.2 OBJETIVOS E JUSTIFICATIVAS

A proposição desse trabalho se justifica pelo fato de que em trabalhos correlatos propostos (LOBO, 2005; OLIVEIRA, 2006; PIRES, 2002; TIMOTEO, 2002), a utilização de algoritmos genéticos para resolução desse problema tem sido feita com relativo sucesso. Com esse trabalho objetiva-se:

- Pesquisar a forma de como se resolve o problema de alocação de horários com a utilização de técnicas da computação evolutiva, mais especificamente os algoritmos genéticos;
- Abordar as restrições que serão utilizadas para a resolução do mesmo;
- Realizar um estudo de algoritmos genéticos e sua utilização para resolução desse tipo de problema;
- Compor os fundamentos teóricos para a construção de um software que resolva esse problema de geração de horários.

1.3 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado da seguinte forma:

O Capítulo 2 apresenta os Algoritmos Genéticos e todos os conceitos envolvidos para construção do modelo proposto.

O Capítulo 3 apresenta a descrição do problema e a apresentação do modelo considerado para sua resolução.

O Capítulo 4 aborda o problema de geração de horários através da utilização de um algoritmo genético.

A conclusão desse trabalho é mostrada no Capítulo 5.

2 ALGORITMOS GENÉTICOS

Este capítulo introduz o conceito de algoritmo genético através de algumas considerações de autores, bem como todos os conceitos envolvidos na utilização de um algoritmo genético (representação cromossômica, população inicial e operadores genéticos).

2.1 DEFINIÇÃO

Goldberg define algoritmo genético como:

[...] um algoritmo de procura baseado nos mecanismos de seleção natural e genética natural. Ele combina a sobrevivência feita por uma função de avaliação entre uma cadeia de caracteres com uma estrutura de informações mudadas aleatoriamente, para formar um algoritmo de procura com algum talento inovador, o mesmo de uma procura de um ser humano. Em toda geração, um novo conjunto de criaturas artificiais (cadeia de caracteres) é criado usando bits e pedaços do teste de avaliação da geração anterior; ocasionalmente uma parte nova é testada. Enquanto aleatório, algoritmos genéticos não são nenhum passeio simples sem destino. A procura mais eficiente das informações anteriores para especular os pontos da nova procura resulta em um aumento na sua performance (apud JUNIOR, 2000, p. 25).

Já Tanomaru conceitua algoritmos genéticos como sendo:

[...] métodos computacionais de busca baseados nos mecanismos de evolução natural e na genética. Em algoritmos genéticos, uma população de possíveis soluções para o problema em questão evolui de acordo com operadores probabilísticos concebidos a partir de metáforas biológicas, de modo que há uma tendência de que, na média, os indivíduos representem soluções cada vez melhores à medida que o processo evolutivo continua (sic) (apud JUNIOR, 2000, p. 25).

Baseando-se nessas conceituações sobre algoritmo genético, podemos inferir que os algoritmos genéticos são algoritmos que se utilizam de uma abordagem evolutiva para a resolução de problemas e que basicamente executam um *loop* para obter uma solução mesmo. Vejamos o fluxo de um algoritmo genético que é mostrado na figura 1:

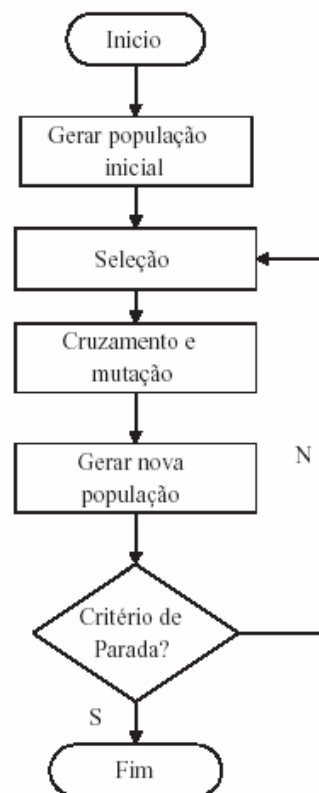


Figura 1: Algoritmo Genético

A figura 1 mostra que um algoritmo genético resolve um problema através da geração, mudança de indivíduos e avaliação das soluções candidatas. A representação de uma solução candidata é denominada cromossomo.

Terminologia:

Indivíduo: representa uma possível solução para a resolução de um problema.

Gene: representa um possível valor para cada posição de um indivíduo.

Cromossomo: representa a forma de representação dessa solução.

Função de aptidão: função a ser maximizada ou minimizada de acordo com o problema proposto. Além disso, cada indivíduo possui um determinado valor, o qual quando decodificado, representa um determinado valor para essa função.

2.2 REPRESENTAÇÃO CROMOSSÔMICA

Para a solução de um problema utilizando algoritmos genéticos, precisamos primeiramente definir uma representação cromossômica, a qual determina como será representada cada possível solução em um problema que utiliza essa técnica evolutiva de resolução. A decisão da representação é muito importante, pois é ela que define o espaço de busca para problema a ser solucionado, e mais, a escolha de uma representação equivocada pode levar a uma solução local para o problema, não à global, que seria a esperada.

As representações mais utilizadas são a binária, a inteira e a de ponto flutuante. A opção por uma ou outra leva em consideração o problema a ser resolvido, e a precisão a ser adotada.

2.3 DEFINIÇÃO DA POPULAÇÃO INICIAL

A definição da população inicial pode ser feita de várias formas. A maneira mais comum é através do início aleatório dos indivíduos (ou candidatos à solução do problema), porém, nessa definição inicial temos que levar em consideração que indivíduos inválidos não podem ser gerados já nessa etapa. Para isso, devemos utilizar algum meio para impedir que esses indivíduos sejam acrescentados a essa população.

A definição do número de indivíduos para a população inicial é variável, e fica realmente definida quando testes são aplicados para verificar a abrangência da população, porém não pode ser muito pequena, pois assim o espaço de exploração das soluções fica muito restrito, e nem muito grande, pois dessa forma a execução

do algoritmo pode se tornar demorada, devido ao aumento do espaço de busca das soluções.

2.4 OPERADORES GENÉTICOS

Os operadores genéticos nada mais são do que critérios baseados na evolução para 'evoluir' os indivíduos de uma população. Os operadores que serão utilizados são a seleção, o cruzamento e a mutação, que serão detalhados abaixo:

2.4.1 Seleção

Esse operador realiza a seleção de determinados indivíduos, de acordo com algum critério, para realizarem a cruzamento posteriormente. Existem diversos critérios de seleção apresentados em trabalhos correlatos, sendo os principais:

Seleção por torneio: nesse método de seleção, dois indivíduos da população são selecionados aleatoriamente, e o indivíduo escolhido é aquele que possui maior ou menor valor na função de aptidão, dependendo se o problema proposto for de maximização ou minimização.

Seleção por elitismo: nesse critério, todos os indivíduos da população são ordenados de acordo com os seus valores na função de aptidão, e os melhores indivíduos são selecionados para passarem para a população seguinte. Esse método garante que os melhores indivíduos não se percam.

2.4.2 Cruzamento

Esse operador, que também pode ser chamado de *crossover*, é um mecanismo de troca de informações genéticas dos indivíduos selecionados para o cruzamento, para a obtenção de novos indivíduos. Isso faz com que mais soluções diferentes sejam exploradas, dentro da gama de soluções possíveis.

Essa recombinação pode ser realizada de várias maneiras, dentre as quais as principais serão apresentadas abaixo:

- **Cruzamento em um ponto:** nesse tipo de recombinação, um ponto nos cromossomos é escolhido, gerando assim dois segmentos em cada cromossomo. Após isso, ocorre a troca de segmentos entre esses cromossomos, conforme veremos abaixo um caso, sendo os cromossomos binários:

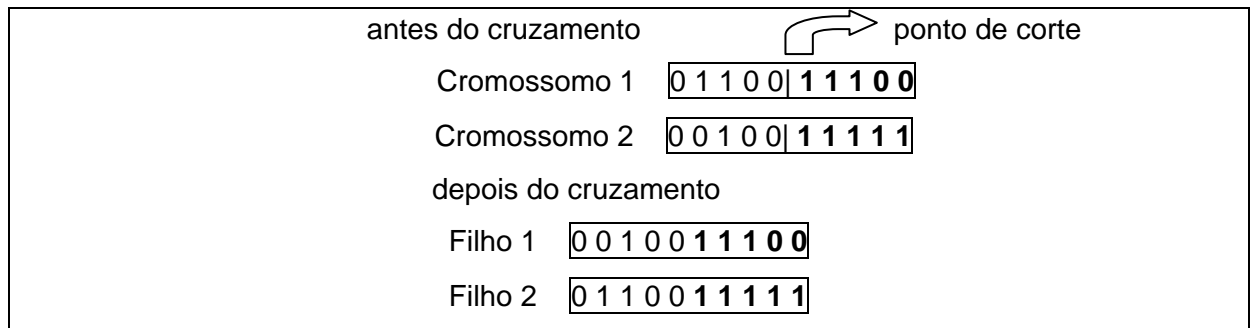


Figura 2: Cruzamento em um ponto

- **Cruzamento em dois pontos:** nesse tipo de cruzamento, dois pontos nos cromossomos são escolhidos, gerando assim três segmentos em cada cromossomo. Após isso, ocorre a troca do segmento do meio entre esses cromossomos, conforme veremos abaixo um exemplo com cromossomos binários:

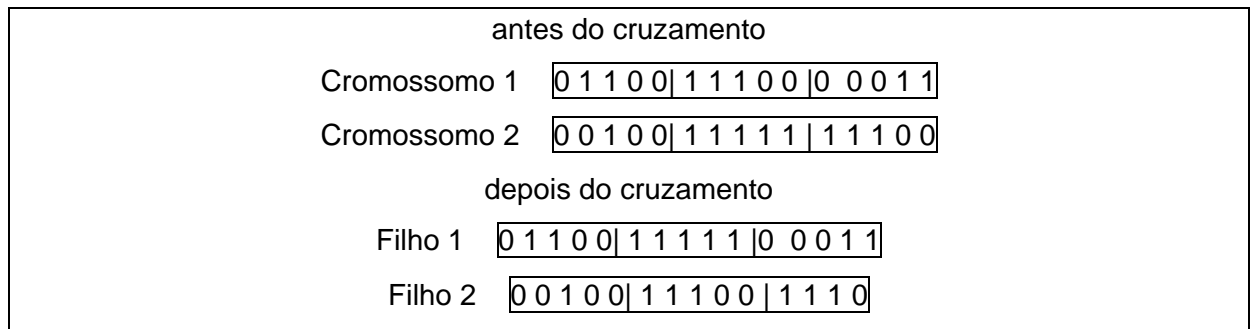


Figura 3: Cruzamento em dois pontos

2.4.3 Mutação

O processo de mutação é uma forma de busca aleatória no cromossomo, sendo que uma posição do mesmo é escolhida de forma eventual, e trocada por um possível valor para esse gene. No caso de um cromossomo com valores binários, caso a posição escolhida para mutação tenha o valor '1', esse valor é mudado para '0'.

Esse operador, de acordo com Cisson (2006) “tem a intenção de prevenir que todas as soluções do problema sejam puramente combinações das soluções da população inicial”, o que aumenta ainda mais a diversidade de soluções. Vejamos uma ilustração desse operador para o caso binário:

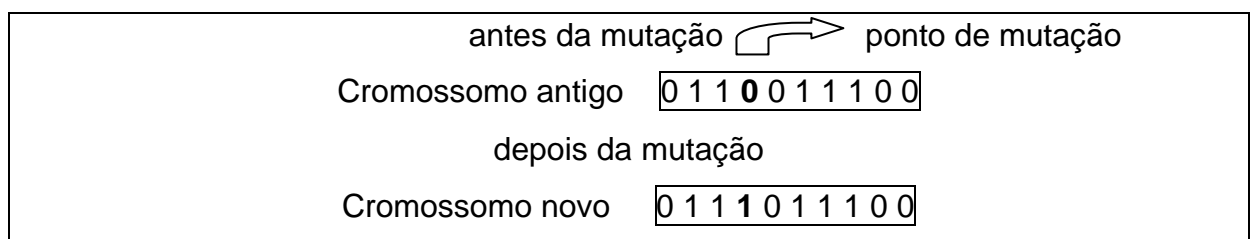


Figura 4: Mutação

2.5 CRITÉRIOS DE PARADA

O Algoritmo Genético necessita de alguma condição de parada, ou seja, que finalize o processo iterativo do algoritmo. Os critérios que podem ser adotados são o tempo, a estagnação da população e o número de gerações

A utilização do tempo pode ser um pouco ruim, visto que, se formos executar o mesmo algoritmo em computadores diferentes e em um mesmo intervalo de tempo, uma máquina pode apresentar uma convergência boa, devido ao seu maior poder de processamento, e a outra pode apresentar uma convergência ruim, devido ao menor poder de processamento dessa máquina.

O critério de estagnação da população também pode ser utilizado, e diz respeito a não melhoria da população após várias gerações do processo iterativo do algoritmo.

Outra condição de término é aplicarmos um número máximo de gerações para a execução do algoritmo.

3 O PROBLEMA DE GERAÇÃO DE HORÁRIOS

Este capítulo mostra o problema de geração de horários considerado, bem como trabalhos correlatos sobre o assunto e o modelo utilizado para resolução do problema, evidenciando todas as restrições consideradas para o horário.

3.1 DESCRIÇÃO DO PROBLEMA

O Problema de Geração de Horários, também descrito na Literatura como *Timetabling Problem*, pode ser definido, de acordo com Wren, como:

[...] o arranjo de horários dentro de padrões de tempo ou espaço, no qual algumas metas são atendidas ou praticamente atendidas e onde restrições devem ser satisfeitas ou praticamente satisfeitas (apud OLIVEIRA, 2006, p. 22).

Esse problema é um tipo amplo de problema, o qual possui algumas classificações, dependendo do tipo de subproblema considerado: *School Timetabling Problem*, *Course Timetabling Problem* ou *Examination Timetabling Problem*.

O *School Timetabling Problem* é um tipo de problema que aborda a alocação de horário em escolas de segundo grau, já o *Course Timetabling Problem* diz respeito à alocação de aulas em uma Universidade e por fim o *Examination Timetabling Problem* aborda a geração de horários de exames em uma Universidade.

3.2 TRABALHOS CORRELATOS

Diversos trabalhos têm sido propostos nessa área de estudo, pois a gama de problemas de geração de horários é NP- Difícil e, devido a isso, diversas técnicas têm sido experimentadas (CISCON, 2006, p. 17). Dentre os métodos utilizados para resolução, podemos citar a busca tabu (SOUSA, 2006, p.29 e JUNIOR, 2000, p. 18), resfriamento simulado (SOARES, 2002, p. 23), redes neurais (FANG, 1994, p. 19), teoria de grafos (JUNIOR, 2000, p. 19) e algoritmos genéticos (TIMOTEO, 2002, p. 29).

Em diversos trabalhos, a escolha dos algoritmos genéticos se dá através de uma comparação com outras técnicas, e através disso, de uma evidenciação das vantagens de sua utilização.

3.3 MODELO CONSIDERADO

O Problema a ser desenvolvido leva em consideração as características de uma Universidade, sendo que o mesmo será proposto para a construção de uma grade horária de somente um curso, mas especificamente o de Ciência da Computação da Universidade Estadual de Londrina.

Para propormos um modelo, precisamos primeiramente de dados para sua construção. Sejam os seguintes dados:

- O curso de Ciência da Computação, com 4 turmas;
- Um conjunto de P Professores;
- Um conjunto S de salas;
- Um conjunto H de horários semanais;
- Cada turma tendo n horários de aulas pré-definidos.

Sejam as seguintes restrições consideradas:

Restrição 1: Um professor não pode ministrar mais de uma aula simultaneamente;

Restrição 2: Impedir mais de duas aulas seguidas da mesma disciplina;

Restrição 3: A turma do 4º ano não pode ter aulas à tarde;

Restrição 4: Todas as turmas não podem ter aulas com professores do Departamento de Computação quarta-feira à tarde, pois os mesmos têm reunião do departamento;

Restrição 5: Os horários de supervisão (TCC e Estágio) não podem coincidir com os horários de aula dos professores supervisores;

Restrição 6: Os horários de supervisão de TCC e Estágio são aulas duplas;

Restrição 7: Duas turmas ou mais não podem ocupar as mesmas salas durante o mesmo horário;

Restrição 8: Horários preferenciais dos professores.

Agora, iremos considerar que um indivíduo é representado pelas quatro turmas e seus respectivos horários. Dessa forma, como estaremos trabalhando com uma população de indivíduos, a solução para o nosso problema será o indivíduo que melhor satisfizer o conjunto de restrições. A função objetivo é:

min. (Restrição 1 + Restrição 2 + Restrição 3 + Restrição 4 + Restrição 5 + Restrição 6 + Restrição 7 + Restrição 8).

4 ABORDAGEM DO PROBLEMA DE GERAÇÃO DE HORÁRIOS POR ALGORITMOS GENÉTICOS

Este capítulo aborda o problema de geração de horários utilizando-se algoritmo genético para sua resolução, evidenciando todas as fases do algoritmo genético.

4.1 REPRESENTAÇÃO DA SOLUÇÃO

Para representarmos uma solução, utilizaremos uma matriz de três dimensões, sendo que as dimensões são o número de turmas (quatro turmas), o número de dias da semana (segunda a sexta – 5 dias) e o número de horários (4 horários diários).

Assim, um indivíduo da população inicial será a matriz de três dimensões descrita acima, mostrada abaixo pelas figuras 5 a 9:

1º ANO				
Segunda	Terça	Quarta	Quinta	Sexta
TECNICAS DE PROGRAMACAO JANDIRA mdc2	TECNICAS DE PROGRAMACAO JANDIR A mdc2	----	CALCULO A NEYVA 155a	COMUNICACAO E EXPRESSAO TEMPORARIO 853
INTRODUCAO CIENCIA COMPUTACAO DIRCEU mdc2	----	CALCULO A NEYVA 853	SISTEMAS DIGITAIS LUCIO 155a	----
SISTEMAS DIGITAIS LUCIO 155a	MATEMATICA DISCRETA SERGIO mdc2	----	----	----
CALCULO A NEYVA 853	TECNICAS DE PROGRAMACAO JANDIR A mdc2	MATEMATICA DISCRETA SERGIO	COMUNICACAO E EXPRESSAO TEMPORARIO 853	COMUNICACAO E EXPRESSAO TEMPORARIO 853

Figura 5: Indivíduo [1]

2º ANO

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS FORMAIS CYNTHIAN m dc1	INGLES SAMANTHA 123	ESTATISTICA STURION	ORGANIZACAO DE COMPUTADORES ELIESER mdc1	ESTATISTICA STURION m dc1
LINGUAGENS FORMAIS CYNTHIAN m dc1	ORGANIZACAO DE COMPUTADORES ELIESER mdc1	INGLES SAMANTHA 123	ORGANIZACAO DE COMPUTADORES ELIESER la b2	ESTRUTURA DE DADOS RODOLFO lab2
INGLES SAMANTHA 12 3	----	----	----	FISICA MARIA TEREZA 484
FISICA MARIA TEREZA 484	----	CALCULO B ELIANDRO 123	CALCULO B ELIANDRO lab1	ESTRUTURA DE DADOS RODOLFO mdc2

Figura 6 Indivíduo [2]

3º ANO

Segunda	Terça	Quarta	Quinta	Sexta
ENGENHARIA DE SOFTWARE RODOLFO lab2	SISTEMAS OPERACIONAIS SAKURAY lab1	ADMINISTRACAO ELIANA 466	ENGENHARIA DE SOFTWARE RODOLFO lab1	COMUNICACAO DE DADOS MARIO lab1
COMUNICACAO DE DADOS MARIO lab1	BANCO DE DADOS VITOR lab1	SISTEMAS OPERACIONAIS SAKURAY mdc2	BANCO DE DADOS VITOR lab1	EMPREENDEDORISMO JANDIRA mdc2
COMPILADORES CYNTHIAN m dc1	COMPILADORES CYNTHIAN ab2	----	COMPILADORES CYNTHIAN lab2	EMPREENDEDORISMO JANDIRA mdc2
ENGENHARIA DE SOFTWARE RODOLFO mdc1	TEORIA LYNEAR mdc1	ADMINISTRACAO ELIANA 466	----	TEORIA LYNEAR lab2

Figura 7: Indivíduo [3]

4º ANO

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	----
COMPUTACAO GRAFICA ALAN lab2	----	----	LINGUAGENS DE PROGRAMACAO KATIA mdc2	I A PEDRO PAULO lab1
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR

Figura 8: Indivíduo [4]

4.2 GERAÇÃO DA POPULAÇÃO INICIAL

A geração da população inicial consiste em se atribuir de maneira aleatória a cada indivíduo da população, ou seja, para todas as turmas, devemos atribuir aleatoriamente dentre os valores possíveis das listas de disciplinas de cada turma.

Nessa geração pode ocorrer de obtermos indivíduos, de determinada turma, com o número de aulas superior ao descrito para essa turma. Devido a isso, devemos adotar um critério que impeça isso, como por exemplo, a exclusão da lista de disciplinas de determinada turma, das disciplinas que já foram escolhidas para compor o horário dessa turma e que já atingiram o número necessário de aulas para essa turma. Assim, podemos evitar que indivíduos que não satisfaçam essa restrição passem adiante para realizar as outras operações genéticas, e garantimos uma melhor qualidade de nossa população inicial.

4.3 SELEÇÃO

Na seleção dos indivíduos selecionamos os melhores indivíduos pelo operador de elitismo, e preenchemos a tabela dos indivíduos restantes para a seleção através do operador de seleção por torneio dos indivíduos que não foram selecionados anteriormente.

Com isso, foram preenchidos metade do número de indivíduos, sendo que a outra metade será obtida através do cruzamento.

4.4 CRUZAMENTO

Já no cruzamento, utilizamos os indivíduos obtidos na seleção para cruzá-los e obtermos novos indivíduos, os quais completarão a população seguinte.

Serão mostrados dois operadores de cruzamento:

- Cruzamento em um ponto;
- Cruzamento em dois pontos.

4.4.1 Cruzamento em um Ponto

Nesse operador, um ponto aleatório é escolhido para ser o ponto do cruzamento, e a partir dele, os genes de dois indivíduos são trocados. Com a troca desses genes entre esses dois indivíduos podemos obter indivíduos novos com uma incompatibilidade na carga horária.

Para impedir a geração desses indivíduos, precisamos trocar apenas os genes compatíveis, ou seja, os genes que representem as mesmas disciplinas. As figuras 9 a 15 representam esse cruzamento.

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	----
COMPUTACAO GRAFICA ALAN lab2	----	----	LINGUAGENS DE PROGRAMACAO KATIA mdc2	I A PEDRO PAULO lab1
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR

Figura 9: Indivíduo A antes do cruzamento

Segunda	Terça	Quarta	Quinta	Sexta
I A PEDRO PAULO lab2	LINGUAGENS DE PROGRAMACAO KATIA lab2	----	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab1
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc2	---	---	COMPUTACAO GRAFICA ALAN mdc2
----	ESTAGIO VITOR	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	ESTAGIO VITOR	TCC LYNEAR
----	ESTAGIO VITOR	----	ESTAGIO VITOR	TCC LYNEAR

Figura 10: Indivíduo B antes do cruzamento

Inicialmente tem-se os indivíduos A (figura 9) e B (figura 10), que passarão pelo cruzamento em um ponto. Após a escolha do ponto de cruzamento, faz-se a intersecção das disciplinas que se repetem nas duas listas, respectivamente dos indivíduos A e B, representada na figura 11.

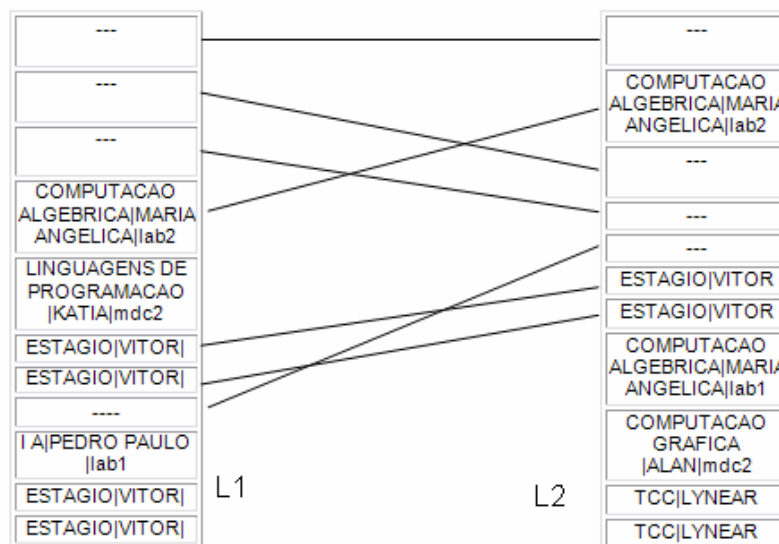


Figura 11: Posições dos indivíduos A e B para serem cruzadas

Com essas duas listas, verificam-se quais disciplinas são iguais nessas listas. Assim, consegue-se obter dois novos indivíduos C (figura 12) e D (figura 13), que ainda estão parcialmente preenchidos.

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	---	
COMPUTACAO GRAFICA ALAN lab2	----	---	---	
TCC LYNEAR	----	COMPUTACAO ALGEBRICA MARIAANGELICA lab2	ESTAGIO VITOR	
TCC LYNEAR	----	---	ESTAGIO VITOR	

Figura 12: Indivíduo C após a troca das disciplinas iguais

Segunda	Terça	Quarta	Quinta	Sexta
I A PEDRO PAULO lab2	LINGUAGENS DE PROGRAMACAO KATIA lab2	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	---
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc2	---		
----	ESTAGIO VITOR	---	ESTAGIO VITOR	
----	ESTAGIO VITOR	---	ESTAGIO VITOR	

Figura 13: Indivíduo D após a troca das disciplinas iguais

Agora, preenchemos o restante de cada indivíduo, a partir do ponto inicial de cruzamento, com as disciplinas de cada lista que não foram selecionadas para o cruzamento.

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	---	LINGUAGENS DE PROGRAMACAO KATIA mdc2
COMPUTACAO GRAFICA ALAN lab2	----	---	---	I A PEDRO PAULO lab1
TCC LYNEAR	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	---	ESTAGIO VITOR	ESTAGIO VITOR

Figura 14: Indivíduo C após o cruzamento

Segunda	Terça	Quarta	Quinta	Sexta
I A PEDRO PAULO lab2	LINGUAGENS DE PROGRAMACAO KATIA lab2	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	---
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc2	---	COMPUTACAO ALGEBRICA MARIA ANGELICA lab1	COMPUTACAO GRAFICA ALAN mdc2
----	ESTAGIO VITOR	---	ESTAGIO VITOR	TCC LYNEAR
----	ESTAGIO VITOR	---	ESTAGIO VITOR	TCC LYNEAR

Figura 15: Indivíduo D após o cruzamento

Com isso, preenchemos os dois indivíduos C (figura 14) e D (figura 15) de forma a se obter os mesmos, porém completos.

4.4.2 Cruzamento em dois Pontos

O processo de cruzamento em dois pontos é idêntico ao cruzamento anterior, diferindo somente pelo fato de que dois pontos são escolhidos para se dar o cruzamento. Sendo assim, o crossover se dá da mesma forma ao cruzamento em um ponto, e é exemplificado nas figuras 16 e 17 antes do cruzamento e pelas figuras 18 e 19 depois do cruzamento.

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	----
COMPUTACAO GRAFICA ALAN lab2	----	----	LINGUAGENS DE PROGRAMACAO KATIA mdc2	I A PEDRO PAULO lab1
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR

Figura 16: Indivíduo A para cruzamento em dois pontos

Segunda	Terça	Quarta	Quinta	Sexta
I A PEDRO PAULO lab2	LINGUAGENS DE PROGRAMACAO KATIA lab2	----	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab1
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc2	---	---	COMPUTACAO GRAFICA ALAN mdc2
----	ESTAGIO VITOR	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	ESTAGIO VITOR	TCC LYNEAR
----	ESTAGIO VITOR	----	ESTAGIO VITOR	TCC LYNEAR

Figura 17: Indivíduo B para cruzamento em dois pontos

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	----	LINGUAGENS DE PROGRAMACAO KATIA mdc2
COMPUTACAO GRAFICA ALAN lab2	----	----	----	I A PEDRO PAULO lab1
TCC LYNEAR	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR

Figura 18: Indivíduo A depois do cruzamento em dois pontos

Segunda	Terça	Quarta	Quinta	Sexta
I A PEDRO PAULO lab2	LINGUAGENS DE PROGRAMACAO KATIA lab2	----	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	----
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc2	---	COMPUTACAO ALGEBRICA MARIA ANGELICA lab1	COMPUTACAO GRAFICA ALAN mdc2
----	ESTAGIO VITOR	----	ESTAGIO VITOR	TCC LYNEAR
----	ESTAGIO VITOR	----	ESTAGIO VITOR	TCC LYNEAR

Figura 19: Indivíduo B depois do cruzamento em dois pontos

4.5 MUTAÇÃO

O operador de mutação consiste na escolha aleatória de duas posições de um indivíduo de uma determinada turma e depois, da troca dessas posições. Essa forma de mutação é feita, pois não podemos simplesmente selecionar uma posição e substituir a disciplina dessa posição, ou uma aula vaga, por outra disciplina qualquer, já que estaremos ferindo a carga horária de aulas já pré-determinada para essa turma.

A figura 11 mostra o indivíduo original A e os genes que foram escolhidos para mutação estão em evidência. Após a mutação obtivemos o indivíduo B, representado na figura 12 com as posições dos genes escolhidos mudadas.

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	----
COMPUTACAO GRAFICA ALAN lab2	----	----	LINGUAGENS DE PROGRAMACAO KATIA mdc2	I A PEDRO PAULO lab1
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR

Figura 19: Indivíduo antes da mutação

Segunda	Terça	Quarta	Quinta	Sexta
LINGUAGENS DE PROGRAMACAO KATIA lab1	I A PEDRO PAULO mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA mdc1	COMPUTACAO ALGEBRICA MARIA ANGELICA lab2	----
COMPUTACAO GRAFICA ALAN lab2	----	----	LINGUAGENS DE PROGRAMACAO KATIA mdc2	----
TCC LYNEAR	----	I A PEDRO PAULO lab1	ESTAGIO VITOR	ESTAGIO VITOR
TCC LYNEAR	----	----	ESTAGIO VITOR	ESTAGIO VITOR

Figura 20: Indivíduo após a mutação

4.6 CRITÉRIO DE PARADA

O critério a ser utilizado é o número de gerações, pois é a iteração do algoritmo através das gerações que produz uma melhora na solução do problema. Outro critério que pode ser utilizado é o tempo de processamento do algoritmo, porém este é determinado pela velocidade de processamento da máquina que estamos utilizando. Sendo assim, a melhor forma de se finalizar o algoritmo é através de um determinado número de gerações decorridas.

5 CONCLUSÃO

Com o desenvolvimento desse trabalho, pôde-se explorar o funcionamento da técnica de otimização heurística - algoritmos genéticos, assim como abordar sua utilização para a resolução de um problema real, que foi a construção de grade de horários.

Em vários trabalhos pesquisados é utilizada a seguinte abordagem: comparação de algoritmos genéticos com outras técnicas de otimização. Não se optou por essa mesma colocação, pois o enfoque do trabalho era primeiramente um estudo dos algoritmos genéticos (população, seleção, mutação, avaliação) de forma mais geral e depois, de forma mais específica, a pesquisa do mesmo para a resolução do Problema de Geração de Horários.

Além disso, esse trabalho serviu como base para a construção de um software de geração de grade de horários para o curso de Ciência da Computação da Universidade Estadual de Londrina.

Como esse trabalho tratou somente o problema e para um determinado curso, pode-se então estendê-lo para diversos cursos.

REFERÊNCIAS

CISCON, L. A. **O PROBLEMA DE GERAÇÃO DE HORÁRIOS: UM FOCO NA ELIMINAÇÃO DE JANELAS E AULAS ISOLADAS**. Monografia (Graduação em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2006. Disponível em: <http://www.comp.ufla.br/monografias/ano2005/O_problema_de_geracao_de_horarios_um_foco_na Eliminacao_de_janelas_e_aulas_isoladas.pdf>. Acesso em: 11 julho 2007.

FANG, H. L. **GENETIC ALGORITHMS IN TIMETABLING AND SCHEDULING**. Tese de PhD, Universidade de Edimburgo, Edimburgo, 1994. Disponível em: <<http://www.cs.unr.edu/~sushil/class/gas/notes/papers/fang94genetic.ps.gz>>. Acesso em: 14 julho 2007.

JÚNIOR, O. O. B. **OTIMIZAÇÃO DE HORÁRIOS EM INSTITUIÇÕES DE ENSINO SUPERIOR ATRAVÉS DE ALGORITMOS GENÉTICOS**. Dissertação (Mestrado em Engenharia de Produção) - Universidade Federal de Santa Catarina. Florianópolis, 2000. Disponível em: <<http://teses.eps.ufsc.br/defesa/pdf/3436.pdf>>. Acesso em: 14 julho 2007.

LOBO, E. L. M. **UMA SOLUÇÃO DO PROBLEMA DE HORÁRIO ESCOLAR VIA ALGORITMO GENÉTICO PARALELO**. Dissertação (Mestrado em Modelagem Matemática e Computacional) - Centro Federal de Educação Tecnológica de Minas Gerais, Belo Horizonte, 2005. Disponível em: <<http://www.mmc.cefetmg.br/info/downloads/D006-EduardoLuizMirandaLobo2005.pdf>>. Acesso em: 12 setembro 2007.

OLIVEIRA, A. C. **USO DO ALGORITMO GENÉTICO E RECOZIMENTO SIMULADO PARA O PROBLEMA DE ALOCAÇÃO DE SALAS**. Monografia (Graduação em Ciência da Computação) – Universidade Federal de Lavras, Lavras, 2006.

PIRES, L. A. **ALGORITMO GENÉTICO APLICADO NA LOCALIZAÇÃO DE ESCOLAS DO MUNICÍPIO DE CORONEL VIVIDA – PR**. Dissertação (Mestrado em Programação Matemática) - Universidade Federal do Paraná, Curitiba, 2002. Disponível em: <<http://ppgmne.cesec.ufpr.br/dissertacoes/liceia.pdf>>. Acesso em: 16 junho 2007.

SOARES, J. A. **A MÁQUINA DE BOLTZMANN NA SOLUÇÃO DO PROBLEMA DE TIMETABLING**. Tese (Doutorado em Engenharia de Sistemas e Computação) - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2002. Disponível em: <<http://www.jsoares.net/artigos/eqJAS.pdf>>. Acessado em: 10 agosto 2007.

SOUSA, V. N. **PROGRAMAÇÃO DA GRADE DE HORÁRIO EM ESCOLAS DE ENSINO FUNDAMENTAL E MÉDIO**. Dissertação (Mestrado em Matemática Aplicada) - Universidade Estadual de Campinas, Campinas, 2006. Disponível em: <<http://libdigi.unicamp.br/document/?code=vtls000382790>>. Acesso em: 10 agosto 2007.

TIMÓTEO, G. T. S. **DESENVOLVIMENTO DE UM ALGORITMO GENÉTICO PARA RESOLUÇÃO DO TIMETABLING**. Monografia (Graduação em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2002. Disponível em: <http://www.comp.ufla.br/monografias/ano2001/Desenvolvimento_de_um_algoritmo_genetico_para_a_resolucao_do_timetabling.pdf>. Acesso em: 11 julho 2007.

WREN, 1996 apud OLIVEIRA, A. C. **USO DO ALGORITMO GENÉTICO E RECOZIMENTO SIMULADO PARA O PROBLEMA DE ALOCAÇÃO DE SALAS**. Monografia (Graduação em Ciência da Computação) - Universidade Federal de Lavras, Lavras, 2006. Disponível em: <http://www.comp.ufla.br/monografias/ano2005/Uso_do_algoritmo_genetico_e_recozimento_simulado_para_o_problema_de_alocacao_de_salas.pdf>. Acesso em: 11 julho 2007.