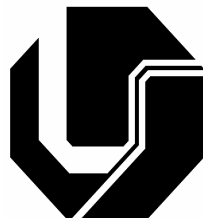


UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELETRICA
POS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



Computação Evolutiva Aplicada ao Problema da
Geração de Grade Horária: o Caso do Curso de
Análise e Desenvolvimento de Sistemas do IFTM

Jairo Gervásio de Freitas
Orientando

Keiji Yamanaka
Orientador

Uberlândia - MG

2012

Jairo Gervásio de Freitas

Computação Evolutiva Aplicada ao Problema da Geração de Grade Horária: o Caso do Curso de Análise e Desenvolvimento de Sistemas do IFTM

Dissertação apresentada por Jairo Gervásio de Freitas à
Universidade Federal de Uberlândia para obtenção do título
de Mestre em Ciências, avaliada em 05 de outubro de 2012
pela banca examinadora

Área de Concentração
Inteligência Artificial

Banca Examinadora
Keiji Yamanaka – UFU (Orientador)
Shiguo Nomura - UFU
Hipólito Barbosa Machado Filho - IFG

Uberlândia - MG

2012

Computação Evolutiva Aplicada ao Problema da Geração de Grade Horária: o Caso do Curso de Análise e Desenvolvimento de Sistemas do IFTM

Dissertação apresentada por Jairo Gervásio de Freitas à
Universidade Federal de Uberlândia para obtenção do título
de Mestre em Ciências, avaliada em 05 de outubro de 2012
pela banca examinadora

Área de Concentração
Inteligência Artificial

Keiji Yamanaka, PhD.
Orientador

Alexandre Cardoso, Dr.
Coordenador do Curso de Pós Graduação

Uberlândia - MG

2012

*Dedico esse trabalho as duas pessoas que mais me
incentivaram e mais acreditaram em mim durante
toda a minha vida: meus pais*

AGRADECIMENTOS:

Agradeço em primeiro lugar a Deus, por ter me dado forças para concretizar mais essa etapa tão importante da minha vida e, por não ter me deixado desanimar nos momentos mais difíceis;

Aos meus pais, por serem meus maiores exemplos e por me ensinarem valores que nenhuma instituição de ensino é capaz de ensinar, como honestidade, simplicidade, perseverança, entre outros;

A minha esposa Thais, pelo amor e companheirismo, por me ajudar muito simplesmente por estar ao meu lado durante todos os momentos desse trabalho;

Ao meu filho Davi, pela enorme ajuda dada em um simples sorriso ou um simples abraço. Sem isso, eu não conseguiria;

As minhas irmãs Marcia, Marisa e Marley, pela torcida a distância;

Aos meus sobrinhos Jéssica e Beto, minhas eternas crianças;

Ao IFTM, pela oportunidade de participar desse curso de mestrado;

Ao meu orientador, Keiji Yamanaka por, apesar das limitações de tempo e distância, sempre contribuir com suas idéias e a sua experiência;

E a todos, que direta e indiretamente, contribuíram para a concretização desse trabalho.

"Algo tão pequeno como o bater de asas de uma borboleta pode causar um tufão do outro lado do mundo"

Teoria do Caos

RESUMO:

A elaboração de grade de horários em instituições de ensino no início de um período letivo é um processo complexo e demorado para os coordenadores de curso ou responsáveis, pois deve atender necessidades de vários envolvidos. A situação se torna ainda pior quando cada professor está disponível para lecionar aulas em apenas alguns períodos da semana. Outro fator que dificulta o processo é que o número de possibilidades de grades horárias é praticamente infinito.

Como o processo de geração de grade de horários é muito particular em cada instituição, já que cada uma possui suas restrições, se torna complicado criar uma solução que atenda todas as instituições.

Para auxiliar esse processo, realizado manualmente em muitas instituições, o presente trabalho tem como objetivo, utilizar o conceito de algoritmos genéticos para criar um sistema que gere em minutos uma grade horária que atenda as particularidades do curso de Análise e Desenvolvimento de Sistemas do Instituto Federal do Triângulo Mineiro, campus Uberaba.

ABSTRACT:

The development of timetable in educational institutions at the beginning of a semester is a complex and lengthy process for course coordinators or responsible, for it must meet various needs of those involved. The situation worsens when each teacher is available to teach classes only in some periods of the week.

As the grid generation process is very specific to each institution, as each has its restrictions, it becomes difficult to create a solution that meets all institutions.

To assist this process, performed manually in many institutions, this work aims, using the concept of genetic algorithms to create a system that generates a timetable in minutes that meets the specific of course of Analysis and Development Systems at the Federal Institute of Triangulo Mineiro, college Uberaba.

SUMÁRIO:

1 – INTRODUÇÃO	15
1.1 – Problema	15
1.2 – Justificativa	17
1.3 – Objetivos do Trabalho	17
1.3.1 – Objetivo Geral.....	17
1.3.2 – Objetivos Específicos	17
1.4 – Estrutura do Trabalho	18
2 – FUNDAMENTOS	19
2.1 – Estrutura de um Algoritmo Genético	21
2.1.1 – Inicialização da população	22
2.1.2 – Cálculo da função de aptidão	23
2.1.3 – Seleção	23
2.1.3.1 – Seleção por Roleta	24
2.1.3.2 – Seleção por Torneio	25
2.1.3.3 – Seleção pelo Método de Amostragem Estocástica Uniforme	25
2.1.3.4 – Seleção Aleatória	25
2.1.4 – Cruzamento	26
2.1.5 – Mutação	28
2.1.6 – Condição de Parada	29
3 – TRABALHOS RELACIONADOS	30
3.1 – Sistema de criação e gerenciamento de grade horária com o método Monte Carlo .	30
3.2 – Sistema Gerador de Grades Horárias – Universidade Regional de Blumenau	31
3.3 – AGHORA	32
3.4 – Times’ cool	33
3.5 – Urânia	34
3.6 – Comparação entre os trabalhos correlatos	35
4 – ARQUITETURA DO SISTEMA	36

4.1 – Cadastro de Dados	37
4.2 – Especificação do Sistema.....	37
4.2.1 – Diagrama de Banco de Dados.....	37
4.2.2 – Diagrama de Casos de Uso	37
4.2.3 – Diagrama de Classes	39
4.2.4 – Diagrama de Atividades.....	39
4.3 – Configuração de Parâmetros	40
4.4 – Algoritmos Genéticos	41
4.4.1 – Estrutura de um Algoritmo Genético.....	41
4.4.2 – Inicialização da população	42
4.4.3 – Cálculo da Aptidão dos Indivíduos.....	43
4.4.4 – Seleção	44
4.4.5 – Cruzamento.....	44
4.4.6 – Mutação	48
4.4.7 – Condição de Parada	49
4.5 – Geração da Grade Horária	49
4.6 – Geração dos Relatórios	54
5 – RESULTADOS E DISCUSSÃO.....	55
5.1 – Estrutura do Curso	55
5.2 – Processo de geração da grade horária	57
5.3 – Análise dos Resultados	58
6 – CONCLUSÃO E TRABALHOS FUTUROS	62
REFERÊNCIAS:	64
ANEXO A: MANUAL DO SISTEMA	66
A.1 – Cadastro de Cursos	66
A.2 – Cadastro de Períodos.....	67
A.3 - Cadastro de Professores.....	68
A.4 – Cadastro de Disciplinas.....	69
A.5 – Relatório completo.....	71

A.6 – Relatório por professores	72
ANEXO B: QUESTIONÁRIO PARA COLETA DE REQUISITOS	74

INDICE DE FIGURAS:

Figura 2.1 – Estrutura de um algoritmo genético canônico	22
Figura 2.2 – Seleção utilizando roleta	25
Figura 2.3 – Cruzamento em um ponto	27
Figura 2.4 – Cruzamento em dois pontos	27
Figura 2.5 – Cruzamento uniforme	28
Figura 2.6 – Mutação	28
Figura 3.1 – Geração de grade horária utilizando o método Monte Carlo.....	31
Figura 3.2 – Geração de grade horária com o sistema da Univ. Regional de Blumenau.....	32
Figura 3.3 – Geração de grade horária utilizando o sistema AGHORA	33
Figura 3.4 – Grade horária de um professor, criada utilizando o sistema Times’cool	34
Figura 3.5 – Geração de grade horária utilizando o sistema Urânia	35
Figura 4.1 – Diagrama de módulos do sistema.....	36
Figura 4.2 – Diagrama do banco de dados.....	37
Figura 4.3 – Diagrama de casos de uso	38
Figura 4.4 – Diagrama de classes	39
Figura 4.5 – Diagrama de atividades	39
Figura 4.6 – Configuração dos parâmetros	41
Figura 4.7 – Representação de uma grade horária.....	42
Figura 4.8 – Representação da população.....	42
Figura 4.9 – Exemplo de cruzamento – Parte 1	45
Figura 4.10 – Exemplo de cruzamento – Parte 2.....	46
Figura 4.11 – Exemplo de cruzamento – Parte 3	46
Figura 4.12 – Exemplo de cruzamento – Parte 4.....	47
Figura 4.13 – Exemplo de cruzamento – Parte 5.....	47
Figura 4.14 – Exemplo de cruzamento – Parte 6.....	48
Figura 4.15 – Exemplo de mutação	48
Figura 4.16 – Tela para geração da grade horária.....	50
Figura 4.17 – Grade horária gerada que não atende todas as restrições obrigatórias	51
Figura 4.18 – Grade horária gerada que atende todas as restrições obrigatórias	52
Figura 4.19 – Grade horária intermediária gerada no sistema	53
Figura 4.20 – Gráficos da maior aptidão encontrada e da média.....	53
Figura 4.21 – Tela de seleção de relatório	54
Figura A.1 – Cadastro de cursos.....	67
Figura A.2 – Cadastro de períodos	68

Figura A.3 – Cadastro de professores	69
Figura A.4 – Cadastro de professores	71
Figura A.5 – Relatório completo da grade horária	72
Figura A.6 – Relatório da grade horária por professores	73

INDICE DE QUADROS:

Quadro 3.1 – Comparação entre alguns sistemas de geração de grade horária	35
Quadro 4.1 – Bônus e penalidades atribuídos na função de aptidão.....	44
Quadro 5.1 – Disciplinas do curso de Análise e Desenvolvimento de Sistemas	56
Quadro 5.2 – Corpo docente do curso de Análise e Desenvolvimento de Sistemas.....	57
Quadro 5.3 – Testes efetuados para encontrar os parâmetros do sistema.....	59

1 – INTRODUÇÃO

1.1 – Problema

O problema da elaboração de uma grade de horários em instituições de ensino é antigo. Todo início de período letivo, os coordenadores de curso gastam muito tempo tentando elaborar uma grade de horários que atenda as necessidades dos professores, a disponibilidade de espaço físico e de recursos.

A situação pode piorar quando existir uma necessidade de uma alteração nos requisitos após todo esse trabalho, nesse caso poderá ser necessário que toda a grade horária que foi gerada tenha que ser alterada.

Segundo Francisco e Silva (2011), em grande parte das instituições de ensino, a geração de grades horárias é realizada manualmente, o que vem a tornar esse processo demorado e sujeito a falhas. Freitas et al. (2007) concordam e acrescentam que o tempo gasto durante o processo é considerável porque é necessário respeitar um grande conjunto de restrições para que o horário obtido satisfaça as necessidades dos envolvidos: os professores e a instituição.

Schaerf (1995) afirma que, na literatura, esse processo de criação de grade horária é conhecido como *Timetabling*, e consiste em determinar uma seqüência de encontros entre estudantes e professores em um espaço de tempo pré-definido (tipicamente uma semana), satisfazendo um conjunto de restrições de vários tipos.

A partir da grande demanda de tempo destinada a essa tarefa, surgiu a necessidade de se automatizar o processo de geração de horários de uma instituição de ensino.

Com o passar dos anos, foram criadas várias alternativas para tentar solucionar esse problema através de sistemas computacionais. Porém, segundo Vieira e Macedo (2011), a automatização de um sistema para este fim é um problema computacional difícil, sem solução algorítmica trivial.

Para Fucilini (2009), tradicionalmente, os professores de uma instituição não possuem todos os seus horários disponíveis para lecionar aulas, seja por que trabalham em outros cursos, na mesma ou em outra instituição ou porque assumem outras atividades. Essa restrição de horários dos professores aumenta a complexidade na busca de uma solução.

Existem várias outras restrições que influenciam esse processo, além da descrita no parágrafo anterior, como a preferência do professor por horários e dias específicos de trabalho, a disponibilidade das salas de aula e laboratórios da instituição, o possível conflito entre o mesmo professor em duas turmas no mesmo horário, dentre outros.

Wang, Liu e Yu (2009) afirmam que as restrições possuem graus de relevância, sendo que restrições de alto grau podem invalidar uma solução, enquanto aquelas restrições de baixo grau não invalidam uma solução, porém, afetam o seu desempenho.

Linden (2012) separa as restrições como *hard constraint* e *soft constraint*. *Hard constraint* são as restrições que obrigatoriamente tem que ser satisfeitas, pois, caso não sejam satisfeitas, fazem com que a solução encontrada, no caso, a grade criada, não possa ser aceita como uma solução para o problema. Por outro lado, *soft constraint* são as restrições desejáveis, ou seja, que visam maximizar o conforto do aluno, professor ou instituição, mas que caso não sejam satisfeitas, não inviabilizam a solução obtida.

De acordo com Linden (2012), o espaço de busca da grade de horários ideal é praticamente infinito, dado que o número de combinações turma/sala/horário é proporcional ao fatorial do número de turmas e salas. Quando impomos restrições, sejam elas obrigatórias ou desejáveis, o problema se complica ainda mais, adquirindo um caráter NP-Completo.

Segundo Cormen (2002), o conjunto dos problemas NP-Completo consiste nos casos em que a complexidade é não polinomial, mas que são verificáveis em tempo polinomial. Isto é, dada uma solução, podemos verificá-la em um tempo razoável.

Quando se analisa um espaço de busca de apenas um período de um determinado curso, com 5 aulas diárias, de segunda a sexta-feira, haveria um total de 25 aulas semanais. 25 aulas semanais geram um espaço de busca equivalente ao fatorial do número 25 que é igual a $1,55 \times 10^{25}$. Ou seja, para um único período se teria $1,55 \times 10^{25}$ possibilidades de grades horárias diferentes.

Se o espaço de busca for ampliado para seis períodos, o número de possibilidades de grades horárias diferentes seria ampliado de maneira exponencial. Haveria um total de 150 aulas semanais gerando um espaço de busca equivalente ao fatorial do número 150, que é igual a $5,71 \times 10^{262}$. Ou seja, para seis períodos se teria $5,71 \times 10^{262}$ possibilidades de grades horárias diferentes.

Com base nos valores apresentados, pode-se dizer que, encontrar uma grade horária que atenda um grande número de restrições dentro de um universo tão grande de possibilidades é uma tarefa complicada de ser realizada.

Segundo Santos (2005), para encontrar a melhor solução (alocação ótima) para este tipo de problema devem-se testar todos os arranjos possíveis, mas isso se torna extremamente complexo quando o número de estados aumenta. Para Linden (2012), problemas cujos algoritmos são extraordinariamente lentos (problemas NP-Completo) ou incapazes de obter solução ideal devem ser tratados através da utilização de algoritmos evolucionários. Os algoritmos genéticos são um ramo dos algoritmos evolucionários e como tal podem ser definidos como uma técnica de busca baseada numa metáfora do processo biológico de evolução natural. Algoritmos genéticos são constantemente utilizados para a solução de problemas relacionados a situações em que, se tem um grande universo de busca no qual se deseja encontrar uma solução que atenda a determinados requisitos.

1.2 – Justificativa

Segundo Vieira e Macedo (2011), existem algumas ferramentas comerciais que prometem a geração automatizada de grades de horários, entretanto, sua utilização é pouco freqüente, uma vez que este tipo de problema envolve necessidades específicas da instituição de ensino.

Nesse contexto, o que ocorre atualmente é que cada instituição de ensino cria a sua própria grade horária, de maneira manual ou automatizada, baseada em suas próprias necessidades, suas características e suas preferências, que são diferentes das encontradas em outras instituições. Sendo assim, se torna complicado desenvolver uma solução computacional genérica que possa atender todas as instituições.

O sistema descrito nesse trabalho analisou as particularidades do curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Ciência, Educação e Tecnologia do Triângulo Mineiro (IFTM), campus Uberaba, onde a grade horária era gerada de maneira manual pelo coordenador de curso.

O curso de Tecnologia em Análise e Desenvolvimento de Sistemas (IFTM, 2009) foi criado no segundo de semestre de 2006, com o objetivo de formar profissionais especializados, empreendedores e pesquisadores, capazes de analisar, projetar, desenvolver e implantar inovações na área de Desenvolvimento de Sistemas.

1.3 – Objetivos do Trabalho

1.3.1 – Objetivo Geral

O objetivo do presente trabalho é desenvolver um *software* que trabalhe com o conceito de algoritmos genéticos e que seja capaz de receber um conjunto de requisitos (períodos, disciplinas e suas quantidades de aulas teóricas e práticas, professores e suas disponibilidades) e a partir deles, aplicar os operadores genéticos (seleção, cruzamento, mutação e elitismo) para gerar uma grade horária para o curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Ciência, Educação e Tecnologia do Triângulo Mineiro, campus Uberaba, que satisfaça todas as restrições obrigatórias informadas e, se possível, todas as restrições desejáveis.

Sendo assim, o objetivo do presente trabalho envolve a solução de um problema do mundo real, atender uma necessidade de uma instituição de ensino, otimizando a tarefa de criação da grade horária. Na instituição de ensino na qual se baseia o estudo de caso, o coordenador de curso gastava horas ou até dias para concretizar essa tarefa, e com o sistema proposto, pretende-se criar a grade horária em poucos minutos.

1.3.2 – Objetivos Específicos

Os objetivos específicos do trabalho são:

- Realizar a pesquisa bibliográfica sobre o tema e sobre outros trabalhos já realizados sobre o assunto;

- Realizar um levantamento de requisitos através de entrevistas com o coordenador do curso de Análise e Desenvolvimento de Sistemas do IFTM - Uberaba;
- Construir um banco de dados com todas as informações necessárias para a construção da grade horária;
- Realizar o levantamento das restrições a serem respeitadas de acordo com as necessidades da instituição;
- Construir um *software* que realize a criação de uma grade de horários de maneira otimizada, utilizando algoritmos genéticos, Delphi como ambiente de programação e Firebird como banco de dados;
- Realizar a demonstração, implantação e testes do *software* com os usuários para verificar o sucesso das restrições já implantadas e detectar novas restrições.

1.4 – Estrutura do Trabalho

Este trabalho está organizado da seguinte forma. Após essa breve introdução, o capítulo 2 apresenta o conceito de algoritmos genéticos, apresentando vantagens da aplicação dessa técnica e explicando os operadores genéticos que podem ser utilizados. O capítulo 3 apresenta *softwares* desenvolvidos por pesquisadores ou empresas para atender a necessidade de geração de grade horária e apresenta também uma comparação entre esses *softwares* e o *software* que está sendo abordado nesse trabalho. O capítulo 4 apresenta a arquitetura utilizada para a construção do sistema e os passos seguidos para a confecção de cada um de seus módulos, bem como a implementação de cada um dos operadores genéticos do Algoritmo Genético. O capítulo também apresenta as funcionalidades de cada uma das telas do sistema, mostrando como são realizadas as operações de cadastro das informações necessárias para o processo, geração da grade horária e obtenção dos relatórios que podem ser impressos. O capítulo 5 apresenta o curso no qual foi aplicado o estudo de caso, mostrando como era realizado o processo de geração de grade horária antes e depois da implantação do sistema. O capítulo também discute os resultados alcançados pelo sistema. O último item que apresentado na dissertação são as conclusões sobre o trabalho realizado e sugestões sobre trabalhos futuros.

2 – FUNDAMENTOS

Uma das áreas da computação que mais atrai a atenção de estudantes e pesquisadores é a Inteligência Artificial. De acordo com Rich (1993), pode-se definir a Inteligência Artificial como a área de pesquisa da Ciência da Computação e da Engenharia da Computação, dedicada a encontrar métodos ou dispositivos computacionais que possuam ou simulem a capacidade racional de resolver problemas, pensar ou, de forma ampla, ser inteligente.

Para Taurion (2009), a Inteligência Computacional é um subconjunto da Inteligência Artificial, e é composta de algoritmos e técnicas que combinam elementos de aprendizado para criar *softwares*, que são, em alguns aspectos, inteligentes. Como exemplo de uso, pode-se citar as Redes Neurais, a Computação Evolucionária e a Lógica *Fuzzy*.

Rezende (2005) fala sobre a Computação Evolutiva que é uma das áreas dentro da Inteligência Computacional que mais se destaca. A Computação Evolutiva ou Evolucionária engloba várias técnicas de busca e otimização inspiradas na genética e evolução natural. A diferença entre a Computação Evolutiva e as técnicas de busca e otimização tradicionais é que, as técnicas de Computação Evolutiva, ao contrário das tradicionais, operam sobre uma população de candidatos em paralelo. Dessa maneira, elas podem realizar a busca em diferentes áreas do espaço de solução, alocando um número de membros apropriado para a busca em várias regiões e podendo obter assim melhores resultados. A Computação Evolutiva se subdivide em algumas áreas como Algoritmos Genéticos, Estratégias de Evolução e Programação Genética.

Segundo Heinen e Osório (2006), os Algoritmos Genéticos tem origem no trabalho do americano John Henry Holland, na década de 60. Eles se baseiam nos conceitos da teoria de evolução das espécies, proposta por Charles Darwin em 1859.

Para Linden (2012), os Algoritmos Genéticos são um ramo dentro da área de Computação Evolutiva e, como tal, usam modelos computacionais dos processos naturais de evolução como uma ferramenta para resolver problemas. Eles funcionam mantendo uma população de estruturas, denominadas indivíduos ou cromossomos, operando sobre estes de forma semelhantes à evolução das espécies, fazendo com que os indivíduos gerados em cada população sejam melhores do que os da população anterior.

O processo de alocação de horário escolar, como o descrito nesse trabalho, é conhecido na literatura como problema de *timetabling*. Freitas et al. (2007), conceituam *timetabling* como problemas complexos de otimização com diversas aplicações práticas, como por exemplo: escalonamento de enfermeiros, horários de aulas e instituições de ensino, planejamento de transporte público e outros.

O processo de alocação de horário escolar é denominado *timetabling* escolar e constitui um problema de otimização combinatória de classe NP-Difíceis, no qual é mais fácil verificar se o resultado apresentado está correto do que obter analiticamente tal solução em tempo polinomial (BARBOZA, 2003 apud FREITAS, 2007).

Para Peranconi (2011), na medida em que aumentam o número de restrições, se torna mais difícil encontrar a solução do problema em tempo hábil, pelos métodos convencionais de programação, surgindo à necessidade de se utilizar metaheurísticas, como os Algoritmos Genéticos (AGs).

De acordo com Lobo (2005), Algoritmos Genéticos são um conjunto de modelos computacionais inspirados na genética. Estes algoritmos modelam uma solução para um problema específico em uma estrutura de dados como a de um cromossomo e aplicam operadores que recombina estas estruturas, preservando informações críticas e descartando informações que não apresentem as características desejadas.

Para Linden (2012), os Algoritmos Genéticos são extremamente eficientes no seu objetivo de varrer o espaço de soluções e encontrar soluções próximas da solução ótima, quase sem necessitar de interferência humana.

Segundo Santos (2005), os Algoritmos Genéticos são programas de computador, que são desenvolvidos para tratar problemas, em geral, de otimização de funções e estruturas ou processos que dependem de um número grande de variáveis, tornando inviável a utilização dos métodos convencionais.

De acordo com Francisco e Silva (2011), dado um problema de otimização qualquer, os Algoritmos Genéticos buscam a resposta a partir de um conjunto aleatório de soluções. Cada uma dessas soluções é chamada de indivíduo ou cromossomo. Um indivíduo representa completamente uma solução para o problema tratado. Os Algoritmos Genéticos favorecem a combinação de indivíduos mais aptos, ou mais promissores à solução de um determinado problema, trabalhando-se com codificação do conjunto de parâmetros e não com os próprios parâmetros.

Os Algoritmos Genéticos são capazes de produzir resultados muito positivos na busca de soluções para problemas complexos, garantindo a obtenção de um resultado que atenda completamente ao problema ou que atenda várias partes dele. Segundo Carvalho (2012), os Algoritmos Genéticos se diferem dos métodos tradicionais de busca e otimização, principalmente em quatro aspectos:

1. AGs trabalham com uma codificação do conjunto de parâmetros e não com os próprios parâmetros.
2. AGs trabalham com uma população e não com um único ponto.
3. AGs utilizam informações de custo ou recompensa e não derivadas ou outro conhecimento auxiliar.
4. AGs utilizam regras de transição probabilísticas e não determinísticas.

De acordo com Lobo (2005), um Algoritmo Genético é um procedimento que mantém uma população de estruturas (chamadas indivíduos), representando possíveis soluções para um determinado problema. Estes indivíduos são avaliados, para gerar oportunidades reprodutivas, de forma que os cromossomos que representam uma solução que melhor atenda ao problema tenham maiores chances de se reproduzir comparados aos que representam uma solução que não tão eficiente. A definição do que seja uma solução que atenda melhor ao problema comparada a outra solução, é tipicamente relacionada à população atual e os critérios para essa análise estão contidos no cálculo da função de aptidão.

Linden (2012) afirma que os Algoritmos Genéticos não constituem um algoritmo de busca da solução ótima de um problema, mas sim uma heurística que encontra boas soluções a cada execução, mas não necessariamente a mesma sempre. Ou seja, a cada execução, um algoritmo genético gera um resultado diferente para o problema.

2.1 – Estrutura de um Algoritmo Genético

Antes de se iniciar a codificação de um Algoritmo Genético, deve-se preocupar com a maneira como será realizada a codificação da informação que se vai trabalhar nos cromossomos. Deve-se modelar o problema que será trabalhado para se encontrar qual é a melhor estrutura para se representar cada cromossomo que será manipulado do AG.

Para Linden (2012), a representação cromossomial é fundamental para um Algoritmo Genético. Basicamente, ela consiste em uma maneira de traduzir a informação do nosso problema em uma maneira viável de ser tratada pelo computador. Quanto mais ela for adequada ao problema, maior a qualidade dos resultados obtidos.

Para Vieira e Macedo (2011), a modelagem do cromossomo é uma etapa de grande importância na formulação de um problema utilizando Algoritmos Genéticos. Uma boa definição do cromossomo minimiza a quantidade de testes necessários para avaliar um indivíduo, além de diminuir o tempo de execução do Algoritmo Genético.

A seguir serão discutidas todas as etapas de um algoritmo genético. A figura 2.1 a seguir mostra a seqüência das operações a serem executadas pelo AG.

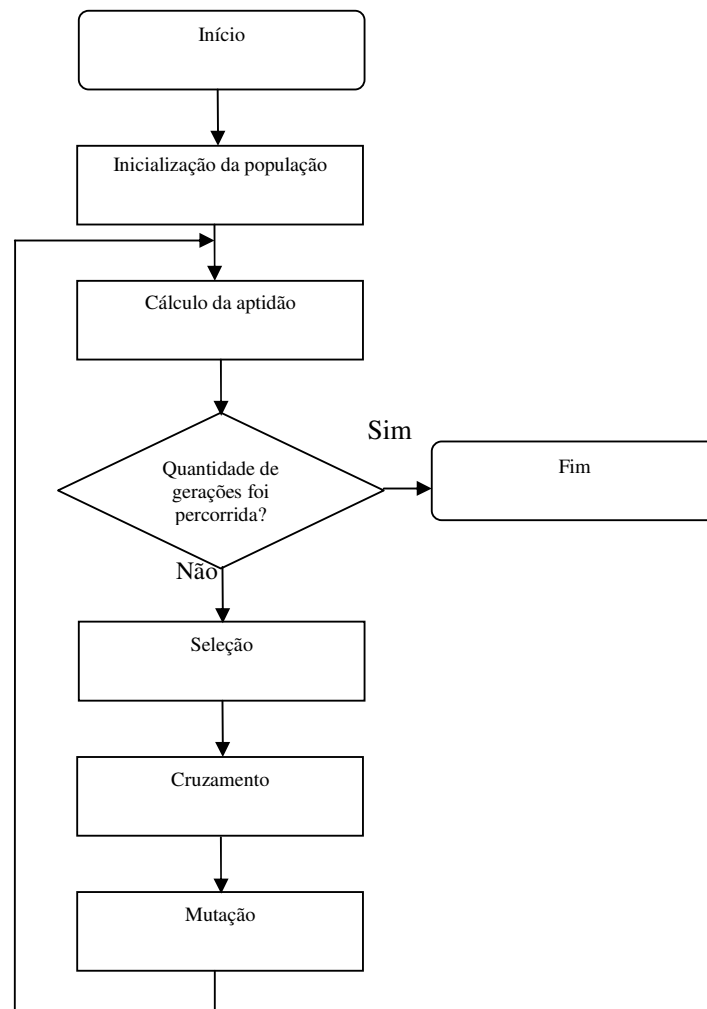


Figura 2.1 – Estrutura de um algoritmo genético canônico – adaptado de Campos (2002)

2.1.1 – Inicialização da população

Na primeira fase do AG, a inicialização da população, são criadas e armazenadas várias alternativas para a solução do problema. Cada uma dessas alternativas recebe o nome de indivíduo. Cada um dos indivíduos criados representa uma possível solução para o problema.

O conjunto de indivíduos criado recebe o nome de população. Antes da execução do Algoritmo Genético se deve determinar quantos indivíduos a população possui.

Segundo Linden (2012), a inicialização da população, na maioria dos trabalhos feitos na área, é feita da forma mais simples possível, fazendo-se uma escolha aleatória independente para cada indivíduo da população inicial. Ou seja, cada indivíduo é criado de maneira que o seu conteúdo seja atribuído aleatoriamente sem preocupação com a solução do problema.

Pode-se destacar também que, nesse momento, foi criada a primeira geração da população. Uma nova geração será criada após os processos de cálculo de aptidão, seleção, cruzamento e mutação.

A quantidade de gerações que será percorrida é outro parâmetro que deve ser configurado antes da execução do AG.

2.1.2 – Cálculo da função de aptidão

Após a geração da população inicial, cada um dos indivíduos é analisado segundo os critérios pré-estabelecidos para verificar o quanto ele atende a solução do problema. Dá-se o nome de função de aptidão ou de função de avaliação à função que verifica a qualidade de um indivíduo como solução do problema proposto.

Segundo Linden (2012), a função de avaliação é a maneira utilizada pelos Algoritmos Genéticos para determinar a qualidade de um indivíduo como solução do problema em questão. A função de avaliação deve, portanto, ser escolhida com grande cuidado. Ela deve embutir todo o conhecimento que se possui sobre o problema a ser resolvido, tanto suas restrições quanto seus objetivos de qualidade.

Linden (2012) também afirma que é essencial que se encontre uma função de avaliação que penalize soluções implausíveis para nosso problema e que avalie satisfatoriamente o grau de adequação de cada indivíduo como solução do problema em questão, de forma que uma solução que confira maior benefício receba uma avaliação mais alta do que aquela que confere um benefício menor.

2.1.3 – Seleção

Para Linden (2012), o método de seleção de pais deve simular o mecanismo de seleção natural que atua sobre as espécies biológicas, em que os pais mais aptos gerem mais filhos, ao mesmo tempo em que permite que os pais menos aptos também gerem descendentes, porém em menor quantidade. Dessa forma, as boas características passam a predominar dentro da nova população de soluções.

Linden (2012) também afirma que os indivíduos menos aptos não devem ser descartados da população reprodutora, pois eles também podem possuir algumas características que sejam boas para a solução do problema.

Realizar a seleção de pais de maneira equivocada pode fazer com que ocorra uma rápida convergência genética de todas as soluções para um mesmo conjunto de características, evitando uma busca mais ampla pelo espaço de soluções. A convergência genética se traduz em uma população com baixa diversidade genética que, por possuir genes similares, não consegue evoluir, a não ser pela ocorrência de mutações aleatórias que sejam positivas.

Existem diversos métodos para a seleção dos indivíduos que participarão dos próximos processos do Algoritmo Genético. A seguir, serão destacados alguns deles. No trabalho proposto, o usuário poderá optar por utilizar a roleta ou o torneio para a seleção dos pais no AG.

2.1.3.1 – Seleção por Roleta

Segundo Bittencourt (2006), a idéia da roleta é que todos os indivíduos de uma população sejam avaliados, e o resultado dessa avaliação seja usado como abertura angular em uma roleta. Sendo assim, indivíduos com maior aptidão teriam um grande ângulo nessa roleta, enquanto indivíduos com menor aptidão teriam ângulos cada vez menores. Com o processo realizado dessa maneira, os indivíduos que possuem um maior ângulo na roleta, conseqüentemente possuem uma maior possibilidade de serem selecionados pela roleta.

Após a roleta estar montada, a seleção dos pais é feita através de um sorteio aleatório de um elemento na roleta.

Lobo (2005) também explica o método de seleção por Roleta, onde indivíduos de uma geração (ou população) são escolhidos para fazer parte da próxima etapa do Algoritmo Genético, através de um sorteio de roleta. Cada indivíduo da população é representado, na roleta, proporcionalmente ao seu índice de aptidão. Dessa forma, para indivíduos com alta aptidão, é dada uma porção maior dessa roleta virtual, enquanto aos indivíduos de aptidão mais baixa é dada uma porção relativamente menor. A roleta é girada um determinado número de vezes, dependente do tamanho da população. A cada giro da roleta, um indivíduo é apontado pela seta e selecionado. Aqueles indivíduos sorteados pela roleta são escolhidos como indivíduos que participarão da próxima etapa do Algoritmo Genético.

A figura 2.2 mostrada a seguir ilustra esse método. Nela, os cromossomos de C1 a C6 estão com a sua aptidão mostrada na coluna da direita. Se somarmos as aptidões dos cinco cromossomos, chegaremos a um valor de aptidão total de 2,2890. Para chegarmos a porcentagem que cada um dos cromossomos possui de ser selecionado pela roleta, deve dividir a sua aptidão pela aptidão total. No caso do primeiro cromossomo (C1), dividi-se 0,5120 por 2,2890, chegando-se a um resultado de pouco mais de 22%. No caso do segundo cromossomo (C2), dividi-se 0,2030 por 2,2890, chegando-se a um resultado pouco menor que 9%. O mesmo raciocínio deve ser aplicado para os outros cromossomos.

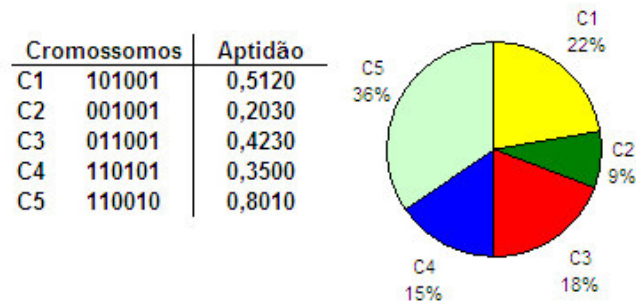


Figura 2.2 – Seleção utilizando roleta – Carvalho (2012)

2.1.3.2 – Seleção por Torneio

De acordo com Linden (2012), o método do torneio consiste em selecionar uma série de indivíduos da população e fazer com que eles entrem em competição direta pelo direito de ser pai, usando como arma a sua avaliação.

Linden (2012) conceitua um parâmetro denominado tamanho do torneio, que define quantos indivíduos são selecionados aleatoriamente dentro da população para competir. Uma vez definido os competidores, aquele dentre deles que possuir a melhor avaliação é selecionado para a aplicação do operador genético.

2.1.3.3 – Seleção pelo Método de Amostragem Estocástica Uniforme

Para Lobo (2005), o método da Amostragem Universal Estocástica ou SUS (*Stochastic Universal Sampling*) pode ser considerado como uma variação do método da roleta, na qual, ao invés de uma única agulha, são colocadas n agulhas igualmente espaçadas, sendo n o número de indivíduos a serem selecionados para a próxima geração. Dessa forma, a roleta é girada uma única vez, ao invés de n vezes, selecionando assim os indivíduos.

2.1.3.4 – Seleção Aleatória

Segundo Freitas et al. (2007), na seleção aleatória, cada indivíduo tem a mesma probabilidade de ser selecionado. Para seleção do indivíduo, é escolhido aleatoriamente um número entre 1 e o tamanho da população. O número selecionado será o número do indivíduo que será utilizado nos próximos passos do Algoritmo Genético.

Nesse método, não existe nenhum critério que faça com que indivíduos mais aptos possam ter maior possibilidade de serem selecionados. Sendo assim, poderá ocorrer a seleção de indivíduos não aptos, dificultando a propagação das características que atendem a solução do problema presentes nos cromossomos.

2.1.4 – Cruzamento

Para Lobo (2005), o princípio básico dos operadores genéticos de cruzamento e mutação 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.

O cruzamento é o processo realizado para que dois indivíduos selecionados no passo anterior possam gerar descendentes, ou seja, novos indivíduos que possam atender melhor a solução do problema. Caso a função de aptidão tenha sido corretamente definida e a seleção tenha sido efetuada de maneira correta, a tendência é que a cada geração, os indivíduos criados sejam melhores que a geração anterior, perpetuando e intensificando as características que tendem a solucionar o problema. Esses princípios são similares aos princípios defendidos pela teoria de Charles Darwin, segundo a qual, a seleção natural privilegia os indivíduos de maior aptidão, com características mais propícias a sobrevivência no meio, conseqüentemente, com maior probabilidade de reprodução e com mais probabilidade de perpetuar seu código genético para as próximas gerações. Essas características mais favoráveis tendem a estar mais presentes nas próximas gerações, enquanto características menos favoráveis tendem a desaparecer aos poucos nas próximas gerações.

Em uma implementação de um AG, para que um cruzamento seja efetuado, é necessário definir uma probabilidade de cruzamento, geralmente determinada através de um número real entre 0 e 1. Nesse caso, para se definir se o cruzamento será efetuado ou não, deve-se sortear aleatoriamente um outro número real entre 0 e 1. Caso esse número seja menor ou igual a possibilidade de cruzamento, este será efetuado. Geralmente se trabalha com um valor alto para a probabilidade de cruzamento. Caso a possibilidade de cruzamento não seja satisfeita, o cruzamento não é efetuado e os indivíduos que foram submetidos ao cruzamento são automaticamente propagados para a próxima geração.

Existem várias maneiras de efetuar o cruzamento de pais e a escolha correta depende da natureza dos dados envolvidos no problema. Por exemplo, se os dados forem binários pode-se optar por criar filhos sendo cópias de um dos seus pais com alguns bits trocados pelos bits do seu outro pai, enquanto o segundo filho seria o contrário do primeiro. Linden (2012) explica um crossover simples, no qual depois de selecionados dois pais pelo módulo de seleção, um ponto de corte é selecionado. Um ponto de corte constitui uma posição entre dois genes de um cromossomo. Depois de sorteado o ponto de corte, os pais são separados em duas partes. O primeiro filho é composto através da concatenação da 1ª parte do primeiro pai com a 2ª parte do segundo pai, enquanto o segundo filho é composto através das partes que sobraram. A figura 2.3 a seguir ilustra esse tipo de cruzamento, chamado de cruzamento em um só ponto.

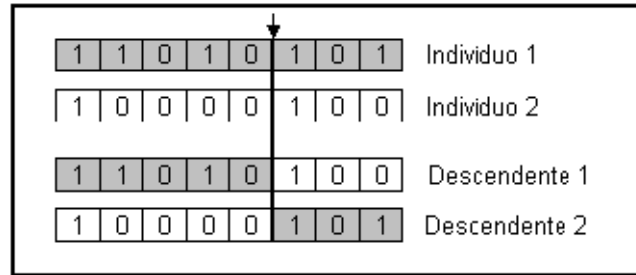


Figura 2.3 – Cruzamento em um ponto – Lobo (2005)

Segundo Linden (2012), para melhorar a capacidade de processar esquemas, podemos utilizar o *crossover* de 2 pontos. Seu funcionamento é similar ao do *crossover* de 1 ponto, com um pequeno acréscimo: em vez de sortearmos um só ponto de corte, são selecionados dois. O primeiro filho é então formado pela parte do primeiro pai fora dos pontos de corte e pela parte do segundo pai entre os pontos de corte e o segundo filho será formado pelas partes restantes. A figura 2.4 a seguir ilustra esse tipo de cruzamento chamado de cruzamento de dois pontos.

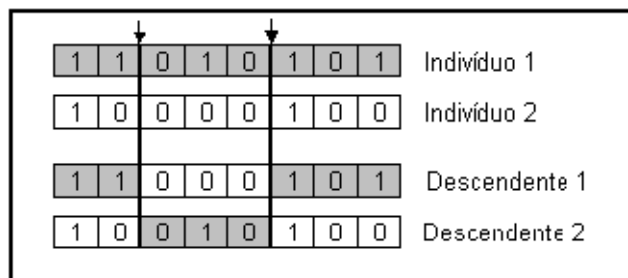


Figura 2.4 – Cruzamento em dois pontos – Lobo (2005)

Linden (2012) afirma que poderia ser necessário aumentar o número de pontos de corte, mas para solucionar esse problema foi desenvolvido o *crossover* uniforme, capaz de combinar todo e qualquer esquema existente. Lobo (2005) define o cruzamento uniforme como um cruzamento significativamente diferente dos outros dois cruzamentos apresentados anteriormente. Conforme ilustrado na figura 2.5 a seguir, primeiramente é criada uma máscara de cruzamento de forma aleatória; posteriormente, cada gene do descendente é criado, copiando-se o gene correspondente de um dos pais, que é escolhido de acordo com a máscara de cruzamento, de modo que, se um certo bit da máscara de cruzamento for 1, o gene correspondente será copiado do primeiro pai; se um certo bit da máscara de cruzamento for 0, será copiado do segundo pai.

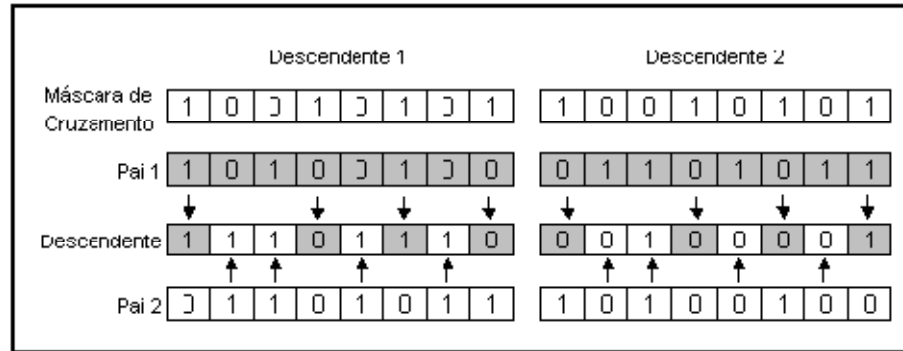


Figura 2.5 – Cruzamento uniforme – Lobo (2005)

2.1.5 – Mutação

A próxima etapa da estrutura de um AG é a mutação. Na implementação da operação de mutação, assim como na operação de cruzamento, deverá ser definido uma probabilidade de mutação, com processo idêntico ao da operação anterior. Mas nesse caso, costuma-se definir um valor baixo para a probabilidade de mutação. Cada um dos indivíduos que passaram pelo cruzamento pode ser submetido à mutação, caso a probabilidade de mutação seja satisfeita. Caso a condição não seja satisfeita, o indivíduo não é alterado.

A mutação ocorre para que alguns filhos possam ter características que não foram herdadas diretamente dos seus pais, garantindo assim uma maior diversidade na população gerada. Para Lobo (2005), a mutação é geralmente vista como um operador de *'background'*, responsável pela introdução e manutenção da diversidade genética na população. Esta operação simplesmente modifica aleatoriamente alguma característica de um ou mais genes do cromossomo 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.

Existem várias maneiras de se efetuar a operação de mutação e a maneira escolhida depende da natureza dos dados do problema. Quando se trabalha com dados binários pode-se optar por inverter o valor de um determinado bit no filho gerado, ou então por trocar o valor de dois bits em posições específicas no filho. A figura 2.6 a seguir exemplifica esse processo. Quando se trabalha com números reais, algumas das alternativas possíveis são gerar novos valores aleatórios dentro de um intervalo específico e substituir pelo valor do filho gerado ou por adicionar ou subtrair um valor aleatório ou não ao valor dele.

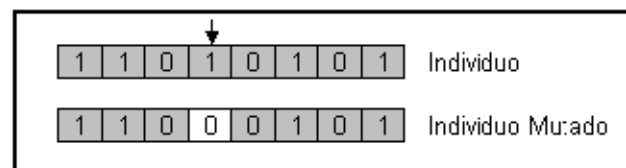


Figura 2.6 – Mutação – Lobo (2005)

2.1.6 – Condição de Parada

Após as etapas de cruzamento e mutação, podem ser utilizadas estratégias para explorar melhor as qualidades da geração atual de forma que elas sejam aproveitadas para a melhoria da próxima geração. Uma dessas possibilidades é o elitismo.

De acordo com Linden (2012), o elitismo é uma pequena modificação no módulo de população que quase não altera o tempo de processamento, mas que garante que o desempenho do AG sempre cresce com o decorrer das gerações.

Para Lobo (2005), o modelo de seleção elitista é utilizado na tentativa de se aumentar a velocidade de convergência do algoritmo, bem como em aplicações nas quais possa ser necessário o seu emprego isoladamente.

O objetivo dessa técnica é garantir que os melhores indivíduos sejam preservados e estejam presentes sempre na próxima geração, garantindo também que esses indivíduos sempre possam gerar descendentes, que podem ser até melhores que os pais. Essa técnica também garante que o indivíduo localizado com melhor aptidão em qualquer geração esteja presente na última geração.

Após serem efetuadas todas as operações, será calculado novamente o valor da aptidão de todos os indivíduos da população da nova geração. Após esse cálculo, poderá ser verificado se algum dos indivíduos gerados atende a solução do problema para que o Algoritmo Genético possa ser encerrado. Não é obrigatório que o algoritmo só seja encerrado quando se encontrar uma solução que atenda 100% ao problema, pode-se optar por encerrar o algoritmo quando, por exemplo, uma determinada quantidade de gerações já tiver sido percorrida. Nesse caso, o indivíduo gerado até aquele momento, que melhor atender ao problema proposto será reconhecido como solução do AG.

Linden (2012) afirma que Algoritmos Genéticos são técnicas probabilísticas, e não técnicas determinísticas. Assim sendo, um Algoritmo Genético com a mesma população inicial e o mesmo conjunto de parâmetros pode encontrar soluções diferentes cada vez que é executado.

3 – TRABALHOS RELACIONADOS

O problema da criação de grade horária é um problema que atrai a atenção de estudantes, pesquisadores e empresas há muito tempo. Existem várias aplicações genéricas que prometem resolver o problema, mas um fato que dificulta essa solução é que cada instituição tem suas próprias necessidades, suas próprias características, suas próprias restrições obrigatórias e desejáveis. Sendo assim, se torna complicado encontrar um *software* genérico que possa atender completamente a realidade de uma instituição de ensino.

Nessa situação, justifica-se que uma instituição de ensino invista na criação do seu próprio *software* que atenda as suas reais necessidades e que possa ser facilmente adaptado, caso surjam novas necessidades ou caso seja necessário alterar alguma das necessidades existentes.

O objetivo desse capítulo é apresentar alguns dos sistemas já desenvolvidos para esse fim e comparar suas funcionalidades com as que são contempladas pelo *software* que está sendo apresentado nessa dissertação.

Existem muitos sistemas que foram desenvolvidos com essa finalidade e o presente trabalho apresenta apenas alguns deles. Esses *softwares* foram escolhidos para serem descritos nessa dissertação por representarem diversas realidades. Foram escolhidos *softwares* que foram desenvolvidos por instituições de ensino e por empresas privadas, que trabalham com Algoritmos Genéticos ou com outras técnicas, que possuem interface para *web* ou *desktop*, que foram desenvolvidas em diversas regiões do país, dentre outras particularidades.

3.1 – Sistema de criação e gerenciamento de grade horária com o método Monte Carlo

Borsoi et al. (2007) desenvolveram um sistema de criação e gerenciamento de grade horária utilizando o método Monte Carlo. O método Monte Carlo é um método estatístico utilizado há bastante tempo como forma de obter aproximações numéricas de funções complexas. Suas aplicações mais comuns são em computação numérica para avaliar integrais. O trabalho foi desenvolvido na Universidade do Vale do Paraíba em São José dos Campos/SP e o seu objetivo era o desenvolvimento de uma aplicação que atendesse a qualquer instituição. Para isso, foi utilizado o Java como linguagem de programação, o XML para armazenamento de valores e da grade horária e o método Monte Carlo para realizar os cálculos para a criação da grade horária.

O sistema possui uma interface sem muitos parâmetros para configuração e de difícil interpretação sobre onde devem ser realizadas as operações. Nele se podem cadastrar matérias, professores e séries. Após esse cadastro, é possível gerar a grade horária correspondente. É permitido cadastrar a disponibilidade ou não de um professor para ministrar aulas em um determinado dia, porém não se pode cadastrar se ele puder apenas em alguns horários do dia. O sistema também não contempla a possibilidade do professor cadastrar horários indesejados, apenas disponíveis e

indisponíveis. Ele também não permite que seja configurado, para que a aula de uma disciplina ocorra obrigatoriamente em um determinado horário. Outra limitação do sistema é que ele não contempla a possibilidade do professor optar por horários simples, duplos ou triplos. A figura 3.1 a seguir mostra uma das telas do sistema, a tela de geração de grades horárias.

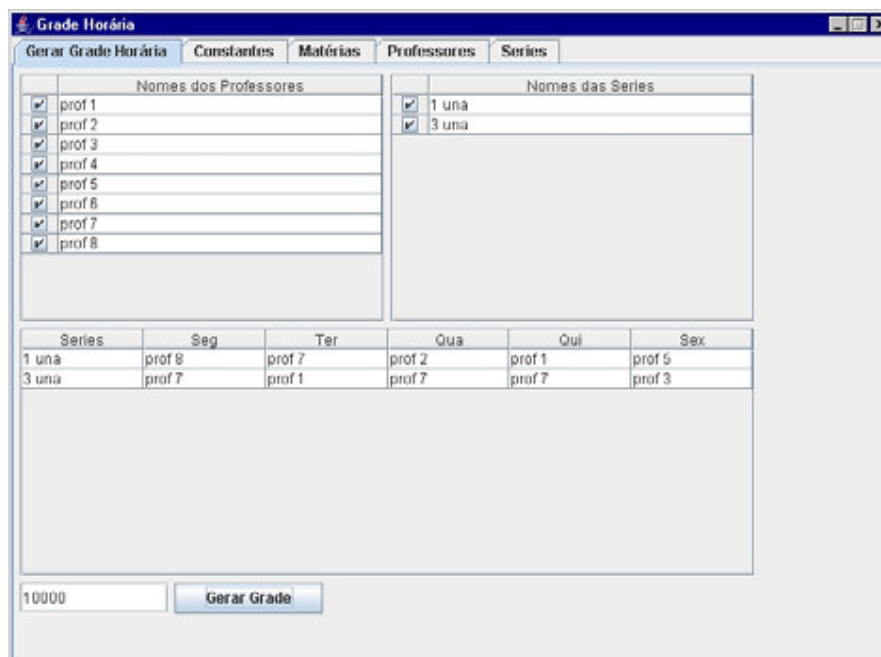


Figura 3.1 – Geração de grade horária utilizando o método Monte Carlo

3.2 – Sistema Gerador de Grades Horárias – Universidade Regional de Blumenau

Preis (2007) apresenta em seu projeto de conclusão do curso de Ciência da Computação, na Universidade Regional de Blumenau em Blumenau/SC, um Sistema Gerador de Grades Horárias que funciona via internet. Na construção do sistema, foi utilizado o Java como linguagem de programação, e o banco de dados MySQL para armazenamento de valores e da grade horária. O sistema trabalha com Algoritmos Genéticos e os seus parâmetros podem ser configurados pelo usuário antes de gerar a grade.

O sistema possui um cadastro de professores, no qual, cada professor é cadastrado e passa a ter um usuário e uma senha para entrar no sistema e cadastrar suas preferências por horários *on-line*. A interface dele é bastante simples para a realização das operações. Algumas de suas limitações é que não é possível cadastrar preferência por ministrar aulas em horários simples, duplos ou triplos e que, um professor só consegue cadastrar seus horários como disponíveis e indisponíveis, não podendo cadastrar seus horários indesejados. Também não é possível colocar um determinado horário como obrigatório para que ocorram aulas de uma determinada disciplina. A figura 3.2 a seguir mostra uma das telas do sistema, no caso, a tela de geração da grade horária.

Curso	Sem.	Disciplina	Professor	Seg	Ter	Qua	Qui	Sex	Sab
BCC (Mat)	1	Fundamentos Matemáticos para Computação	Ticiano Aurélio Campigotto	3,4	3,4				
BCC (Mat)	1	Introdução à Computação	Paulo Roberto Dias				1,2		
BCC (Mat)	3	Arquitetura de Computadores	Antonio Carlos Tavares		1	4		1,2	
BCC (Mat)	3	Cálculo Diferencial e Integral VII	Neide de Melo Aguiar Silva	3,4				4,5	
BCC (Mat)	3	Estruturas de Dados I	Paulo César Rodacki Gomes		3,4	1,2			
BCC (Mat)	3	Física Experimental	Elcio Schumacher	2	5			3	
BCC (Mat)	3	Programação II	José Roque Voltolini da Silva		2		3,4,5		
BCC (Mat)	3	Teoria da Computação	José Roque Voltolini da Silva	1		3,5	1		

Concluído

Figura 3.2 – Geração de grade horária com o sistema da Univ. Regional de Blumenau

3.3 – AGHORA

O sistema AGHORA (Algoritmos Genéticos para Geração de Horários de Aula) foi desenvolvido na UNAMA (Universidade da Amazônia) por Júnior e Rocha (2005).

Ele é composto por um módulo de gerência de informações acadêmicas e um módulo de geração de horário. No módulo de gerência de informações acadêmicas, são cadastrados professores, cursos, disciplinas, turmas e departamentos. Os professores têm sua disponibilidade cadastrada como disponível, indisponível ou indesejável, para cada um dos horários de cada dia da semana, porém não se pode cadastrar sua preferência por horários simples, duplos ou triplos. Também não é possível cadastrar para que uma determinada disciplina tenha aulas obrigatoriamente em um determinado horário. No módulo de geração de horário devem ser informados os parâmetros para a execução do Algoritmo Genético e é nele que a grade horária é gerada e, caso se deseje, impressa. Outra limitação do sistema é o alto tempo que é gasto para que a grade horária final seja criada. A figura 3.3 a seguir mostra uma das telas do sistema, no caso, a tela de geração da grade horária.

Professor	Disciplina	Horário
Miguel	Álgebra Linear e Geometria Analítica	1
Ricardo	Algoritmo e Programação de Computadores	2
Miguel	Álgebra Linear e Geometria Analítica	5
Edson	Física e Circuitos Digitais	6
Neves	Lógica e Matemática Discreta	7
Edson	Física e Circuitos Digitais	8
Felix	Teoria dos Processos Organizacionais	10
Aleksandro	Introdução à Computação	11
Ricardo	Algoritmo e Programação de Computadores	12
Rúbia	Inglês Inter. Aplicado à Computação	13
Rúbia	Metodologia de Pesquisa	14
Neves	Lógica e Matemática Discreta	15

Tempo Gasto = 02:30:34

Imprimir Sair

Figura 3.3 – Geração de grade horária utilizando o sistema AGHORA

3.4 – Times'cool

O Times'cool, lançado em fevereiro de 2007, é um sistema para geração de horários que funciona via internet e pode ser utilizado gratuitamente. Para isso, basta realizar um cadastro no site, informando nome de usuário, nome da escola, e-mail, senha, dentre outros. Não é informada qual é a tecnologia utilizada durante o processo.

Depois de realizada essa etapa, o usuário cadastra turmas, professores, matérias e depois atribui aulas a cada uma das turmas. Após o cadastro, pode-se gerar a grade horária e imprimir a grade gerada por professor, por turma ou da escola. O próprio site onde o sistema é acessado possui um tutorial que mostra para o usuário como devem ser realizadas as operações.

Uma funcionalidade que se destaca no sistema é que é possível cadastrar prioridades entre os professores para que uns tenham maior prioridade para satisfazer suas restrições em relação a outros. Os professores também têm sua disponibilidade cadastrada como disponível ou indisponível. Por outro lado, o sistema não permite que sejam cadastrados horários indesejados pelo professor, nem permite que se opte por gerar horários simples, duplos ou triplos. Também não é possível cadastrar um horário obrigatório para uma determinada disciplina. Outro fator negativo é que a interface gráfica é bastante limitada. A figura 3.4 a seguir mostra uma das telas do sistema, no caso, a tela com a grade horária contendo as aulas que serão ministradas por um professor.

The screenshot shows a web browser window with the URL 'www.lotimiza.com' and the logo 'Times'cool'. Below the browser window is a table representing a weekly class schedule. The table has 6 columns: 'Prof(a)' (with a calendar icon and the name 'Jo'), 'S', 'T', 'Q', 'Q', and 'S'. The rows are numbered 1 to 4. The cells contain course codes like 'Mat 5A', 'Mat 3B', and 'Mat 1D'. Some cells are shaded red, indicating unavailability or conflicts.

Prof(a) "Jo"	S	T	Q	Q	S
1	Mat 5A	Mat 5A	Mat 3B	Mat 3B	
2	Mat 3B	Mat 5A	Mat 3B	Mat 5A	
3	Mat 1D	Mat 5A	Mat 3B	Mat 1D	
4	Mat 1D	Mat 1D		Mat 1D	

Figura 3.4 – Grade horária de um professor, criada utilizando o sistema Times'cool

3.5 – Urânia

No ponto de vista empresarial, um dos *softwares* mais consolidados no mercado é o Urânia, desenvolvido pela GEHA Sistemas Especialistas de Curitiba/PR, desde 1986. O Urânia atende mais de 6000 usuários espalhados por todo o território brasileiro. Não é informada qual é a tecnologia utilizada durante o processo de geração de grade horária.

Algumas das principais características do sistema são a possibilidade de determinar em quais horários o professor está disponível, possibilidade de trabalhar com aulas individuais ou germinadas, possibilidade de trabalhar com divisão de turmas para dois professores, possibilidade de limitar o número de aulas diárias de cada professor, possibilidade de minimizar a ocorrência de janelas, possibilidade de colocar uma disciplina como obrigatória em um determinado horário semanal, possibilidade de determinar o número máximo de aulas por dia para uma disciplina, dentre muitas outras. O Urânia também se destaca pela grande quantidade de possibilidades que podem ser configuradas para melhor atender as necessidades específicas de cada instituição, além da grande quantidade de relatórios possíveis de serem impressos após a geração da grade. Por outro lado, uma das limitações do sistema é que se trata de um *software* pago, ou seja, para que a instituição possa continuar utilizando o *software* é necessário renovar a licença periodicamente. Em contrapartida, a instituição que adquire o *software* tem direito a suporte técnico gratuito por telefone ou *e-mail*, além de ter acesso a novas versões do produto desenvolvidas durante o período de contrato. Outra limitação é que o *software* também não permite que sejam cadastrados horários indesejados pelo professor, ou seja, horários onde ele pode ministrar aulas, porém não quer. A figura 3.5 a seguir mostra uma grade horária gerada utilizando o Urânia.

Turma 5*1*							Turma 5*2*						
Hor	Seg	Ter	Qua	Qui	Sex	Sáb	Hor	Seg	Ter	Qua	Qui	Sex	Sáb
07:10	EdFis	Geo	Ingl	EdFis	Mat	-----	07:10	Port	Mat	Port	Geo	Port	-----
08:00	Mat	Cie	Cie	Hist	Cie	-----	08:00	Port	Mat	Mat	EdFis	Port	-----
08:50	Port	Port	Hist	Mat	Port	-----	08:50	Inf	Cie	Mat	Ingl	EdArt	-----
10:00	Inf	Port	Mat	Geo	EdFis	-----	10:00	EdFis	Hist	EnsR	Hist	Cie	-----
10:50	Port	EdArt	EnsR	Mat	Ingl	-----	10:50	Cie	Geo	Ingl	Mat	EdFis	-----
11:40	-----	-----	-----	-----	-----	-----	11:40	-----	-----	-----	-----	-----	-----

Turma 6*1*							Turma 6*2*						
Hor	Seg	Ter	Qua	Qui	Sex	Sáb	Hor	Seg	Ter	Qua	Qui	Sex	Sáb
07:10	Cie	Port	EnsR	Mat	EdFis	-----	07:10	Mat	Hist	Mat	Ingl	EdArt	-----
08:00	Hist	Hist	Port	Geo	EdArt	-----	08:00	EdFis	Port	EnsR	Inf	Mat	-----
08:50	EdFis	Mat	Port	EdFis	Ingl	-----	08:50	Cie	Geo	Mat	Geo	EdFis	-----
10:00	Port	Geo	Mat	Mat	Cie	-----	10:00	Cie	Cie	Port	Mat	Port	-----
10:50	Inf	Port	Mat	Ingl	Cie	-----	10:50	Hist	Ingl	Port	EdFis	Port	-----
11:40	-----	-----	-----	-----	-----	-----	11:40	-----	-----	-----	-----	-----	-----

Figura 3.5 – Geração de grade horária utilizando o sistema Urânia

3.6 – Comparação entre os trabalhos correlatos

O quadro 3.1, mostrado a seguir, apresenta uma comparação entre os *softwares* correlatos descritos nesse capítulo e o *software* desenvolvido para o IFTM – Campus Uberaba, de acordo com alguns critérios.

Critérios	Método Monte Carlo	Univ. Regional de Blumenau	AGHORA	Times'cool	Urânia	Software do IFTM
Permite cadastrar a disponibilidade do professor para cada horário da semana	Não	Sim	Sim	Sim	Sim	Sim
Permite cadastrar a disponibilidade do professor para cada horário como desejada, indesejada ou indisponível	Não	Não	Sim	Não	Não	Sim
Permite que o professor possa optar por horários simples, duplos ou triplos para cada disciplina	Não	Não	Não	Não	Não	Sim
Permite que sejam cadastrados horários obrigatórios para disciplina	Não	Não	Não	Não	Sim	Sim
Software gratuito	Sim	Sim	Sim	Sim	Não	Sim

Quadro 3.1 – Comparação entre alguns sistemas de geração de grade horária

4 – ARQUITETURA DO SISTEMA

É importante destacar que antes de se iniciar o desenvolvimento do sistema, foram realizadas várias conversas com o coordenador do curso de Análise e Desenvolvimento de Sistemas (ADS) do Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro (IFTM), campus Uberaba, para levantamento de todos os requisitos para criação do *software*. Foram recolhidas as prioridades da instituição e as necessidades da coordenação de curso em relação a um *software* para geração de grade horária.

É essencial obter corretamente todos esses requisitos antes de se iniciar o desenvolvimento do sistema para que haja economia de tempo e agilidade no desenvolvimento do *software*. Por outro lado, caso os requisitos não forem corretamente levantados e o desenvolvimento do sistema se iniciar, caso seja verificado que é necessário realizar alguma alteração nos mesmos, será perdido muito tempo até que sejam efetuadas as correções.

O sistema gerenciador de banco de dados escolhido para se trabalhar no sistema de geração de grade horária foi o Firebird 1.5. As informações que foram cadastradas no mesmo são informações reais e foram obtidas através dessas conversas com o coordenador do curso de Análise e Desenvolvimento de Sistemas.

O sistema foi desenvolvido utilizando, como ambiente de programação, o Borland Delphi 7. O Delphi 7 foi escolhido por ser um ambiente que possibilita o desenvolvimento de sistemas, com grande variedade de componentes gráficos e grande facilidade de interação entre esses componentes e o código fonte do sistema e o sistema gerenciador de banco de dados. Os módulos desenvolvidos estão mostrados na figura 4.1 a seguir.

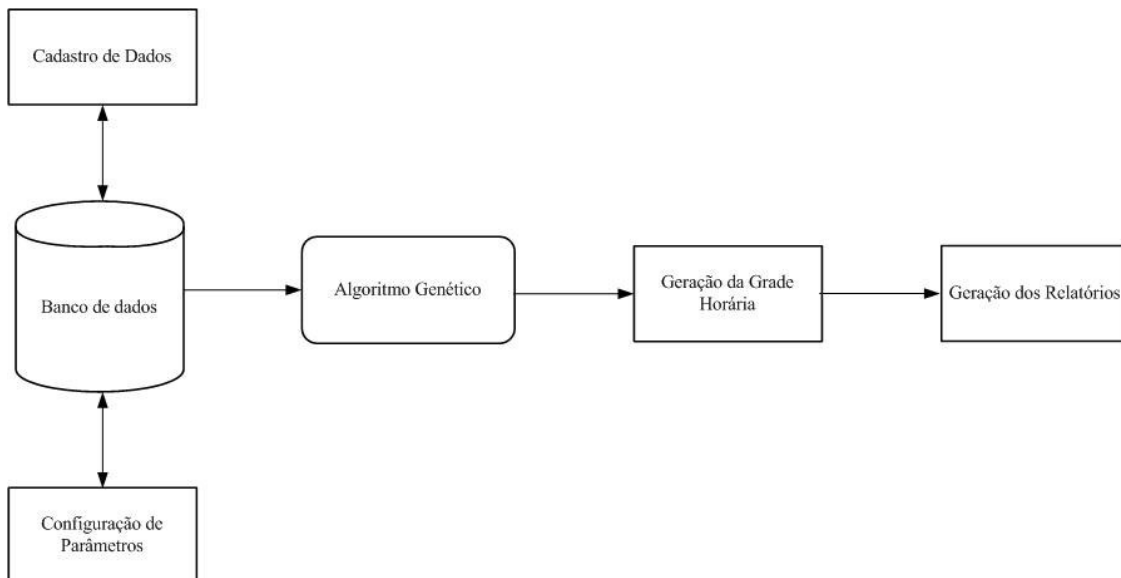


Figura 4.1 – Diagrama de módulos do sistema

4.1 – Cadastro de Dados

O módulo de cadastro de dados será utilizado pelo coordenador do curso para a realização das operações de cadastro das informações referentes ao curso de Análise e Desenvolvimento de Sistemas (ADS), que serão utilizadas durante o processo de geração da grade horária.

Esse módulo será descrito no ANEXO A deste trabalho

4.2 – Especificação do Sistema

Os tópicos subseqüentes irão mostrar a especificação do sistema, mostrando alguns dos seus diagramas.

4.2.1 – Diagrama de Banco de Dados

O banco de dados utilizado para a geração da grade horária possui quatro tabelas: tb_cursos, tb_periodos, tb_professores e tb_disciplinas. A figura 4.2 a seguir mostra o diagrama do banco de dados utilizado, com as tabelas, seus campos e seus respectivos relacionamentos.

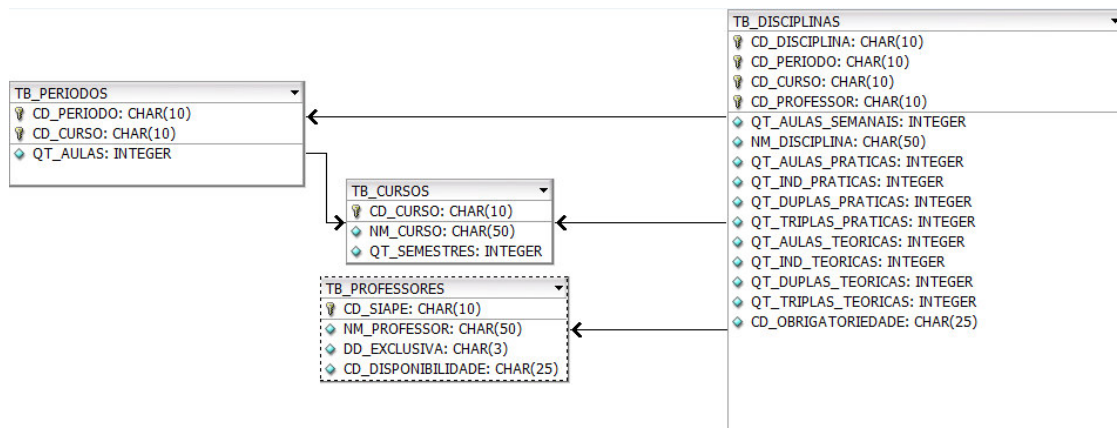


Figura 4.2 – Diagrama do banco de dados

4.2.2 – Diagrama de Casos de Uso

A figura 4.3 a seguir representa os casos de uso do sistema.

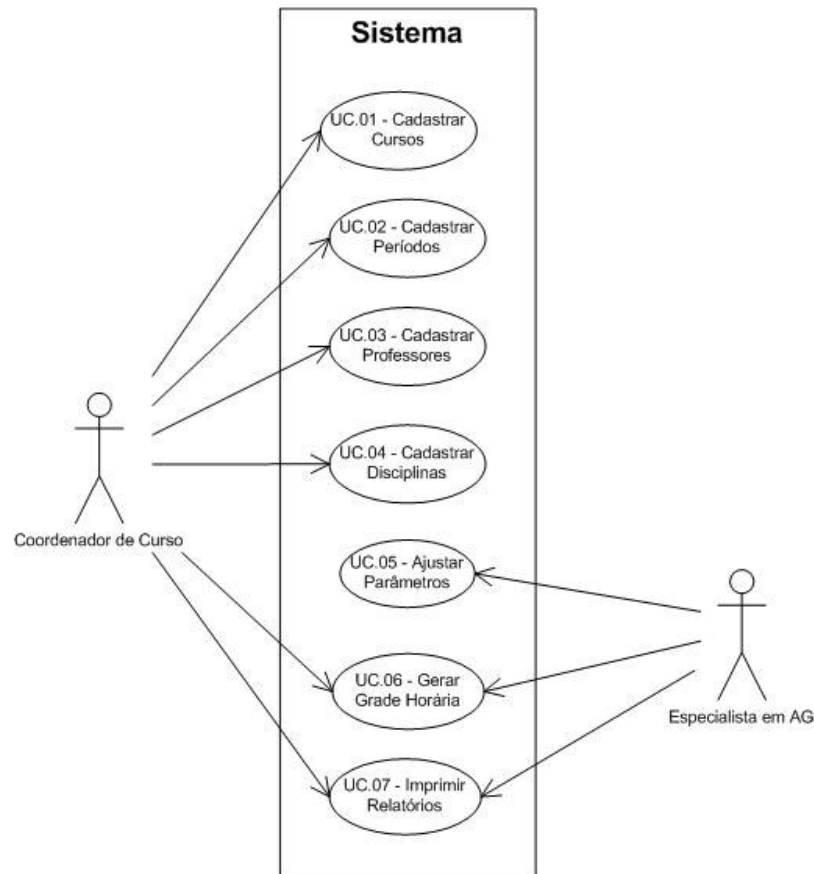


Figura 4.3 – Diagrama de casos de uso

A seguir são descritas as principais características dos casos de uso:

- UC.01 – Cadastrar Cursos – através deste caso de uso é possível cadastrar, editar ou remover cursos cadastrados no sistema;
- UC.02 – Cadastrar Períodos – neste caso de uso é possível cadastrar, editar ou remover períodos cadastrados para cursos disponíveis no sistema;
- UC.03 – Cadastrar Professores – através deste caso de uso é possível cadastrar, alterar informações ou remover professores no sistema;
- UC.04 – Cadastrar Disciplinas – neste caso de uso é possível cadastrar, editar informações ou remover disciplinas no sistema;
- UC.05 – Ajustar Parâmetros – através deste caso de uso é possível alterar os parâmetros de execução do AG;
- UC.06 – Gerar Grade Horária – este caso de uso permite a geração da grade horária do curso;
- UC.07 – Imprimir Relatórios – através deste caso de uso é possível imprimir os relatórios referentes as grades horárias geradas pelo sistema.

4.2.3 – Diagrama de Classes

A figura 4.4 a seguir mostra o diagrama de classes utilizado no sistema.

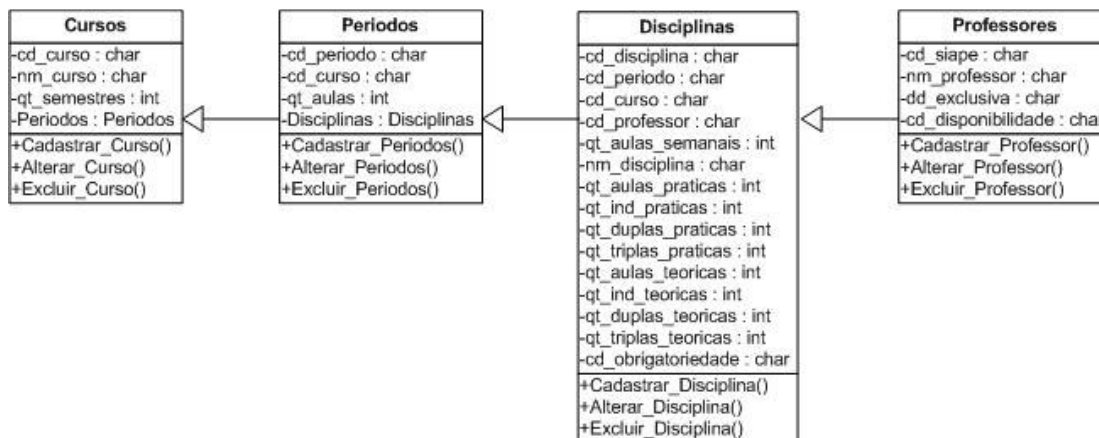


Figura 4.4 – Diagrama de classes

4.2.4 – Diagrama de Atividades

A figura 4.5 a seguir representa o diagrama de atividades que indica o fluxo executado para a geração das grades horárias.

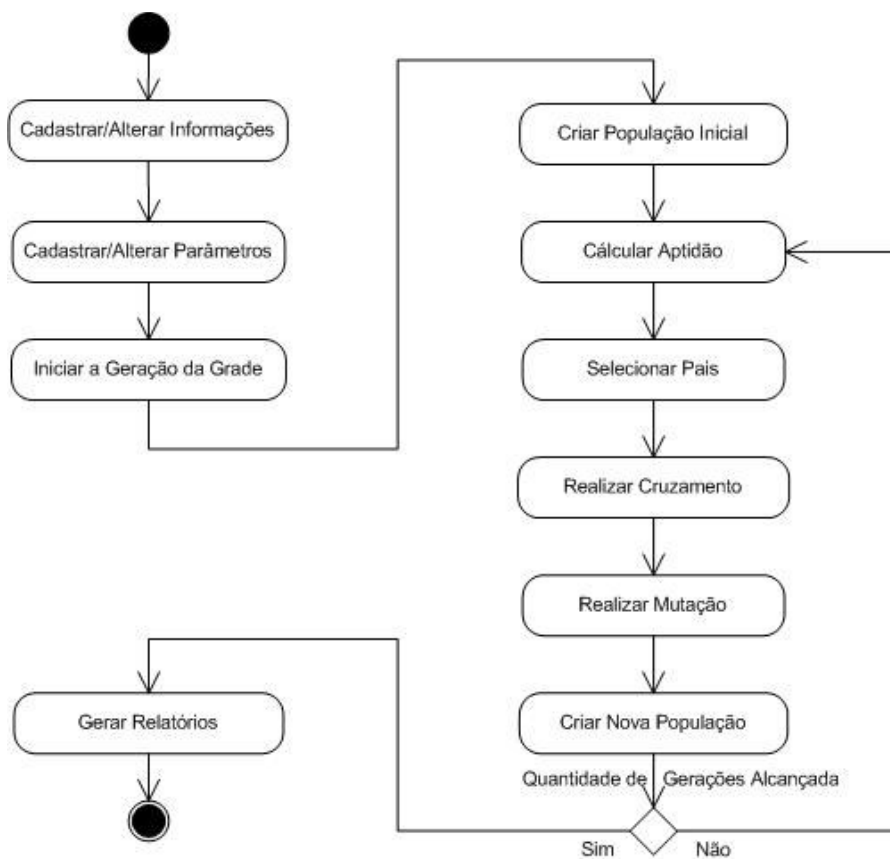


Figura 4.5 – Diagrama de atividades

A atividade “Cadastrar/Alterar Informações” é executada pelo coordenador de curso e se refere à operação adicionar ou alterar os cursos, períodos, professores e disciplinas cadastrados no sistema, caso haja necessidade. A atividade “Cadastrar/Alterar Parâmetros” é executada pelo especialista em Algoritmos Genéticos e se refere à operação adicionar ou alterar os parâmetros do sistema. A atividade “Iniciar a Geração da Grade”, como o próprio nome diz, representa o início da execução do sistema. A atividade “Gerar Relatórios” oferece a possibilidade da geração e impressão dos relatórios referentes à grade horária. As demais atividades são executadas pelo Algoritmo Genético.

4.3 – Configuração de Parâmetros

O módulo de configuração de parâmetros é o módulo onde deverão ser configurados os parâmetros que serão utilizados pelo Algoritmo Genético, como:

- Quantidade de gerações a serem percorridas;
- Quantidade de indivíduos de cada geração;
- Taxa de cruzamento;
- Taxa de mutação;
- Tipo de cruzamento: um ou dois pontos;
- Método de seleção dos pais: roleta ou torneio.

Para acessá-la, basta clicar em Operações e depois em Alterar Parâmetros. Essa tela deve ficar sob responsabilidade de uma pessoa com conhecimento em Algoritmos Genéticos, pois pode haver necessidade que esses parâmetros tenham que ser reajustados para que o sistema consiga encontrar uma solução que atenda o problema ou para que o tempo para que a solução seja encontrada seja reduzido.

Caso seja realizado um ajuste dos parâmetros, os novos valores podem ser salvos para utilização em uma nova execução do sistema. Para isso, basta clicar no botão Salvar.

A figura 4.6 a seguir mostra a tela para configuração dos parâmetros do Algoritmo Genético. Essa tela também possui opção para teste dos parâmetros configurados. Essa opção será mostrada detalhadamente no item 4.6.

Ajuste de Parâmetros para a Geração da Grade de Horários

Número de Gerações:

População (máximo de 5000):

Taxa de Cruzamento (entre 0 e 1):

Taxa de Mutação (entre 0 e 1):

Tipo de Cruzamento:

Um Ponto

Dois Pontos

Método de Seleção dos Pais:

Roleta

Torneio

Figura 4.6 – Configuração dos parâmetros

4.4 – Algoritmos Genéticos

Esse módulo é o responsável por receber e processar as informações cadastradas pelo usuário para, a partir delas, gerar a grade horária. Nesse item, serão descritas como foram executadas cada uma das funções do Algoritmo Genético.

4.4.1 – Estrutura de um Algoritmo Genético

Definir a maneira como será realizada a codificação da informação que será trabalhada nos cromossomos é essencial para o desempenho do Algoritmo Genético. Deve-se modelar o problema que será trabalhado para se encontrar qual é a melhor estrutura para se representar cada cromossomo que será manipulado pelo AG. Quanto mais a representação for adequada ao problema, maior a qualidade dos resultados obtidos.

Para se escolher a estrutura para armazenamento da informação foram realizados vários estudos e optou-se por representar a grade horária de cada indivíduo em uma estrutura do tipo matriz, com duas dimensões. A matriz foi escolhida por ser uma estrutura de dados de fácil manipulação, além de apresentar boa velocidade de acesso às suas informações. Uma dimensão da matriz indica o período do curso e a outra, indica o horário semanal. Os horários semanais são numerados de 1 a 25 e cada algarismo indica a posição da matriz na qual o horário é representado. O número 1 o primeiro horário de segunda-feira, o número 2 o segundo horário de segunda-feira e assim respectivamente, até o número 25 que representa o último horário de sexta-feira. Em cada posição da matriz, representada por um horário semanal e um período do curso, é armazenado o código da disciplina que deve ser ministrada naquele período naquele horário. O código da disciplina, por sua vez, é representado por um grupo de caracteres alfanuméricos entre 2 e 5 elementos e foi definido pela coordenação de curso como identificador único da disciplina. A figura 4.7 mostrada a seguir exemplifica a representação de um indivíduo.

Período	Segunda					...	Sexta				
1°	ALG	ALG	ALG	PDI	PDI	...	PRO	PRO	PRO	SI	SI
2°	ARQ	ARQ	ARQ	FM	FM	...	EST	EST	EST	MOD	MOD
3°	POO	POO	PES	BD	BD	...	SO	SO	SO	MD	MD
4°	RED	RED	RED	PV	PV	...	PPI	PPI	PPI	MF	MF
5°	SD	SD	DAW	DAW	DAW	...	PDM	PDM	PDM	AOE	AOE
6°	SEG	SEG	SEG	COM	COM	...	GPS	GPS	COM	NTC	NTC

Figura 4.7 – Representação de uma grade horária

Quando se fala de toda a população de indivíduos, pode-se dizer que ela é representada por uma matriz de três dimensões, na qual, a terceira dimensão equivale ao número do indivíduo da população. A figura 4.8 a seguir mostra a representação da população do Algoritmo Genético, nesse caso são mostrados apenas três indivíduos.

Período	Segunda					...	Sexta								
1°	PDI	PDI	PDI	SI	SI	...	ALG	ALG	ALG	PRO	PRO				
2°	EST	Período	Segunda					...	Sexta						
3°	PES	1°	PRO	PRO	PRO	ING	ING	...	ALG	ALG	ALG	PDI	PDI		
4°	RED	2°	MOD	Período	Segunda					...	Sexta				
5°	DAW	3°	SO	1°	ALG	ALG	ALG	PDI	PDI	...	PRO	PRO	PRO	SI	SI
6°	COM	4°	PPI	2°	ARQ	ARQ	ARQ	FM	FM	...	EST	EST	EST	MOD	MOD
		5°	PDM	3°	POO	POO	PES	BD	BD	...	SO	SO	SO	MD	MD
		6°	NTC	4°	RED	RED	RED	PV	PV	...	PPI	PPI	PPI	MF	MF
				5°	SD	SD	DAW	DAW	DAW	...	PDM	PDM	PDM	AOE	AOE
				6°	SEG	SEG	SEG	COM	COM	...	GPS	GPS	COM	NTC	NTC

Figura 4.8 – Representação da população

4.4.2 – Inicialização da população

A inicialização da população consiste em gerar uma quantidade de grades horárias de acordo com os parâmetros do sistema. Cada grade horária da população inicial representa um indivíduo, uma possível solução para o problema.

Para a geração de cada grade horária é selecionada individualmente cada disciplina de cada período do curso. Com a disciplina selecionada, é escolhido aleatoriamente um número entre 1 e 25 que representa a posição da grade horária na qual será inserida a disciplina. Caso essa posição já esteja ocupada por outra disciplina, outra posição é selecionada até que se encontre um local disponível. Quando se encontra uma posição disponível, a inserção é realizada. A quantidade de vezes que uma mesma disciplina é inserida na grade é determinada pela quantidade de aulas semanais que ela possui. Quando essa quantidade for atingida, a próxima disciplina será selecionada.

Pode-se destacar que todos os indivíduos gerados representam grades horárias válidas, ou seja, grades horárias que foram geradas de acordo com as informações cadastradas no banco de dados do sistema. Dessa maneira, todos os períodos das grades horárias geradas possuem as disciplinas

corretas, com a quantidade de aulas correta de cada uma delas. Nenhuma grade horária é gerada com uma disciplina localizada em um período incorreto do curso ou com a quantidade de aulas semanais diferente da quantidade cadastrada.

Nesse momento, ainda não são consideradas as preferências de cada professor. Essas preferências serão abordadas durante o cálculo da aptidão dos indivíduos fazendo com que, a cada geração, o Algoritmo Genético se encarregue de encontrar uma grade horária que atenda melhor as necessidades dos envolvidos no processo. Caso as preferências de cada professor fossem consideradas durante a geração da população inicial, o mérito da solução do problema não seria exclusivamente do Algoritmo Genético.

Para a geração de cada um dos indivíduos da população inicial também é considerada a questão das aulas obrigatórias de serem ministradas em algum horário específico. Sendo assim, caso alguma das disciplinas tenha horários obrigatórios teóricos ou práticos cadastrados, esses horários estarão presentes na posição correta em cada uma das grades geradas.

4.4.3 – Cálculo da Aptidão dos Indivíduos

Para se efetuar o cálculo da aptidão dos indivíduos é levado em conta o cadastro realizado previamente no sistema. Nele, para cada professor foi cadastrada sua disponibilidade de ministrar aula em cada um dos horários semanais, sua preferência por ministrar aulas individuais, duplas ou triplas para cada disciplina, a obrigatoriedade de uma disciplina ter que ser ministrada em um horário específico, entre outros.

Para se chegar ao valor da aptidão de cada indivíduo, se trabalha com o somatório de valores a título de bônus e penalidade a um valor de aptidão inicial. O valor de aptidão inicial é o mesmo para todos os indivíduos e é necessário para que, o valor da aptidão nunca seja negativo após somarem-se os bônus e as penalidades do indivíduo. Com base nos testes realizados, chegou-se a conclusão de que o valor da aptidão inicial recomendado é de 70000.

Quando se atende uma restrição, se ganha um valor de bônus, que é somado ao valor da aptidão de um indivíduo. Quando não se atende uma restrição, tem-se uma penalidade, que é subtraída do valor da aptidão.

Para se realizar o cálculo do valor da aptidão do indivíduo, se utiliza a seguinte equação:

$$f(x) = \text{Aptidão inicial} + \sum(\text{bônus}) - \sum(\text{penalidades})$$

Os valores dos bônus e penalidades são definidos de acordo com o nível de importância associado ao cumprimento ou não cumprimento daquela restrição. As restrições obrigatórias quando são violadas acarretam um alto valor de penalidade, enquanto as restrições desejáveis quando são violadas acarretam um menor valor de penalidade. Alguns dos bônus e penalidades contemplados pelo sistema e seus respectivos pesos estão mostrados no quadro 4.1 a seguir.

Restrição	Bônus / Penalidade	Peso
Professor indisponível no horário	Penalidade	1000
Professor dar aula em dois lugares ao mesmo tempo	Penalidade	1000
Disciplina ministrada em horários não consecutivos	Penalidade	10
Disciplina ministrada em mais de 3 horários por dia	Penalidade	10
Professor não quer ministrar aula no horário	Penalidade	10
Ocorrência de janelas	Penalidade	1
Professor dar aula em horário duplo ou triplo, quando indicado	Bônus	10
Professor ministrar várias aulas no mesmo dia	Bônus	10
Professor dar aulas em horários disponíveis	Bônus	1

Quadro 4.1 – Bônus e penalidades atribuídos na função de aptidão

4.4.4 – Seleção

O sistema possibilita a utilização de mais de um tipo de seleção para os pais que passarão pelos operadores genéticos. Na tela de configuração de parâmetros, pode-se optar pela utilização dos métodos de seleção via roleta ou via torneio.

Quando se for utilizar a roleta, deve-se ressaltar a necessidade de se somar uma constante ao valor da aptidão de todos os indivíduos para evitar que seja possível encontrar um valor de aptidão negativo para um indivíduo. Caso esse valor não seja somado, um indivíduo poderia receber um valor maior de penalidade do que o valor de bônus, tendo então um valor negativo de aptidão. Esse fato faria com que a roleta fosse montada de maneira errada, fazendo com que não houvesse a proporcionalidade correta para cada espaço da roleta, comprometendo todo o restante do Algoritmo Genético.

Quando se trabalha com o torneio, deve-se destacar que são selecionados aleatoriamente três indivíduos da população para a disputa de cada lugar para se submeter aos operadores genéticos. Primeiramente, são selecionados três indivíduos e o indivíduo que tiver o maior valor de aptidão dentre eles, será o escolhido para ser o primeiro indivíduo que irá para a próxima etapa do Algoritmo Genético. Depois são selecionados novamente três indivíduos e o indivíduo que tiver o maior valor de aptidão dentre eles, será o escolhido para ser o segundo indivíduo que seguirá para as próximas etapas do processo.

4.4.5 – Cruzamento

Na etapa de cruzamento, os dois pais selecionados na etapa anterior são submetidos ao operador genético do cruzamento, caso seja satisfeita a condição da taxa de cruzamento escolhida pelo usuário na tela de parâmetros.

Na tela de configuração dos parâmetros do Algoritmo Genético, pode-se optar por utilizar o cruzamento por um ou por dois pontos. Independente do método escolhido, o processo para cruzamento é bastante similar. Nele, os dois indivíduos que passarão pelo processo de cruzamento são chamados de pais, enquanto os dois indivíduos gerados pelo cruzamento são chamados de filhos.

No cruzamento em um só ponto, um ponto aleatório é escolhido como ponto de cruzamento. Trabalha-se com cada período letivo individualmente. As disciplinas do primeiro pai do primeiro ponto até o ponto de cruzamento são copiadas do primeiro pai para o primeiro filho. As demais disciplinas a partir desse ponto serão copiadas do segundo pai. Esse processo é efetuado para cada período. Para a formação do segundo filho, executa-se o mesmo processo apenas substituindo o primeiro pai pelo segundo pai e vice-versa.

No cruzamento de dois pontos, dois pontos aleatórios são selecionados e serão trocadas as disciplinas entre esses dois pontos em cada período. O primeiro filho será composto pela parte entre os dois pontos de cruzamento do segundo pai e pelo restante do primeiro pai. O segundo filho será composto pela parte entre os dois pontos do primeiro pai e as demais partes do segundo pai. Esse processo é executado para cada um dos períodos da grade horária.

Neste tipo de cruzamento, deve-se preocupar com a maneira como essa troca de disciplinas entre as grades será efetuada, pois caso contrário, poderá ser gerada uma grade horária incompatível, ou seja, com uma quantidade de aulas de cada disciplina diferente da que está cadastrada no sistema.

A figura 4.9 a seguir mostra a representação das disciplinas de um período para dois pais que irão se submeter a um cruzamento de dois pontos e parte dos dois filhos com o conteúdo já atribuído. Caso o primeiro filho fosse preenchido pela parte entre os dois pontos de cruzamento do segundo pai e pelas demais partes do primeiro pai seria gerada uma grade horária incompatível com as informações das disciplinas. Isso ocorreria porque os dois pais possuem, por exemplo, 3 aulas semanais da disciplina F e o primeiro filho iria possuir 6 aulas semanais da disciplina, enquanto o segundo filho não iria possuir nenhuma aula da disciplina. Problemas similares iriam ocorrer com outras disciplinas tornando a grade horária gerada incompatível.

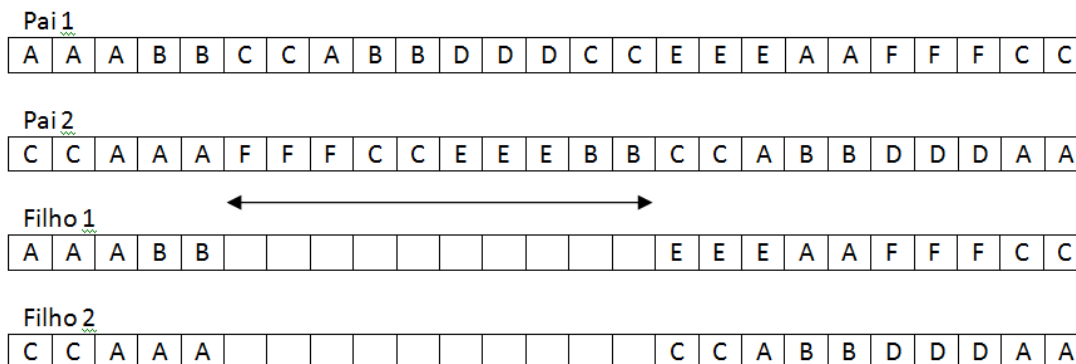


Figura 4.9 – Exemplo de cruzamento – Parte 1

Para impedir a geração dessas grades horárias incompatíveis, devem ser trocadas de local apenas as disciplinas que estiverem presentes nos dois pais, dentro do intervalo de cruzamento. Ou seja, deve ser selecionada a primeira disciplina após o ponto de cruzamento no primeiro pai. Deve-se procurar se essa disciplina está presente no segundo pai em uma posição após o ponto de cruzamento dentro do intervalo de cruzamento, caso esteja, ela estará localizada em uma posição X do segundo pai. Essa disciplina deverá então ser copiada para o primeiro filho para a posição X. Esse processo de busca das disciplinas do primeiro pai no segundo pai deve ser efetuado até a última posição do primeiro pai, se for um cruzamento de um só ponto ou até o segundo ponto de cruzamento se for um cruzamento de dois pontos.

As figuras 4.10 a 4.14 a seguir ilustram esse cruzamento. Na figura 4.10, com a disciplina C selecionada no primeiro pai e assinalada com a cor azul deve-se procurar se ela está presente no segundo pai dentro do intervalo de cruzamento. Como ela está presente no segundo pai, foi também assinalada com a cor azul e o seu conteúdo deve ser copiado para o primeiro filho na posição que a disciplina ocupa no segundo pai.

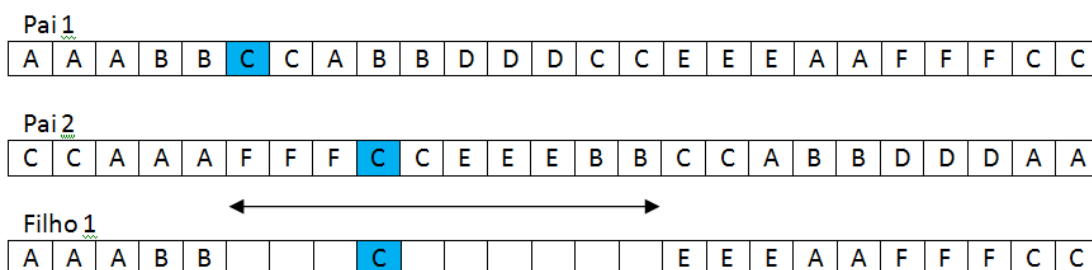


Figura 4.10 – Exemplo de cruzamento – Parte 2

Na figura 4.11 a seguir, é analisada a próxima posição do intervalo de cruzamento no primeiro pai. Essa posição também possui como conteúdo a disciplina C e foi assinalada com a cor azul no primeiro pai. Deve-se procurar se ela está presente no segundo pai dentro do intervalo de cruzamento. Nesse caso, ela está presente na posição assinalada com a cor vermelha no segundo pai. Como essa posição já foi utilizada, procura-se uma nova posição. A próxima ocorrência da disciplina dentro do intervalo de cruzamento do segundo pai está assinalada com a cor azul. O seu conteúdo deve ser copiado para o primeiro filho na posição que a disciplina ocupa no segundo pai.

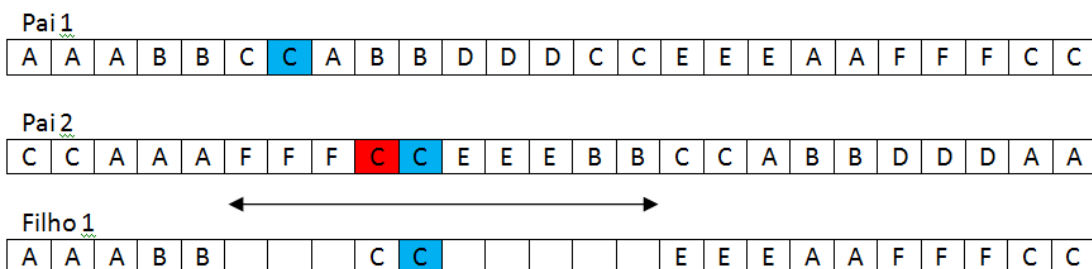


Figura 4.11 – Exemplo de cruzamento – Parte 3

Na figura 4.12 a seguir, é analisada a próxima posição do intervalo de cruzamento no primeiro pai. Essa posição possui como conteúdo a disciplina A e foi assinalada com a cor azul no primeiro pai. No segundo pai estão assinaladas com a cor vermelha, as disciplinas que já tiveram o seu conteúdo copiado para o primeiro filho. Deve-se procurar se a disciplina A está presente no segundo pai dentro do intervalo de cruzamento. Nesse caso, ela não está presente e, nesse momento, não deve ser copiada para o primeiro filho. O seu conteúdo é copiado para uma estrutura chamada de posições faltantes que armazena as disciplinas que não tiveram o seu conteúdo encontrado no segundo pai dentro do intervalo de cruzamento. Essas disciplinas serão utilizadas posteriormente.

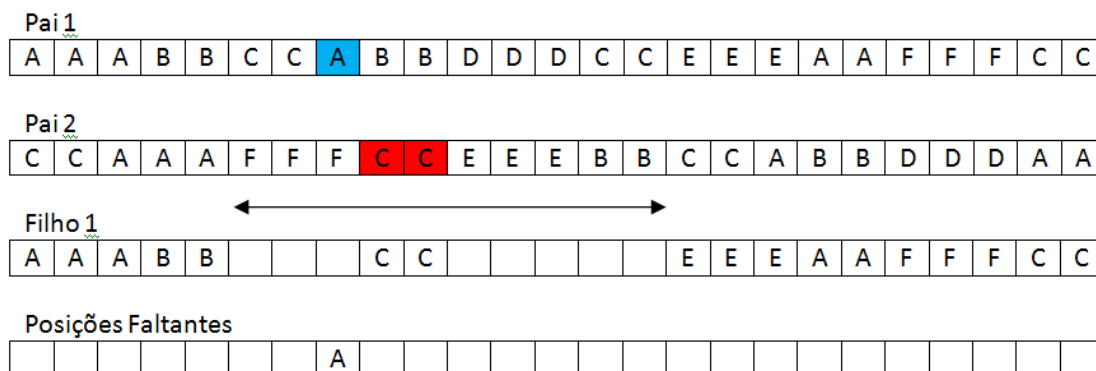


Figura 4.12 – Exemplo de cruzamento – Parte 4

A figura 4.13 a seguir mostra o primeiro filho após todas as disciplinas do primeiro pai dentro do intervalo de cruzamento tiverem o seu conteúdo buscado no segundo pai, segundo a estratégia adotada. Estão assinaladas com a cor vermelha no segundo pai, as disciplinas que foram encontradas e tiveram o seu conteúdo copiado para o primeiro filho na posição que ocupavam no segundo pai. As posições que não foram encontradas estão armazenadas na estrutura destinada as posições faltantes.

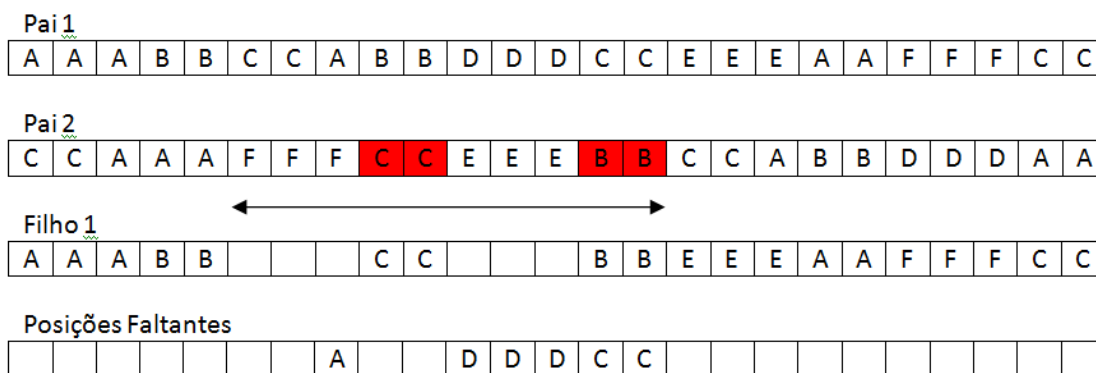


Figura 4.13 – Exemplo de cruzamento – Parte 5

Após esse passo, devem-se preencher as posições restantes, ou seja, preencher as disciplinas presentes no primeiro pai no intervalo de cruzamento, mas que não estão presentes no segundo pai nesse mesmo intervalo. Essas disciplinas serão copiadas para o primeiro filho sequencialmente em

cada espaço que ainda não recebeu nenhuma disciplina. A figura 4.14 a seguir ilustra esse procedimento mostrando o conteúdo do primeiro filho após essa etapa.

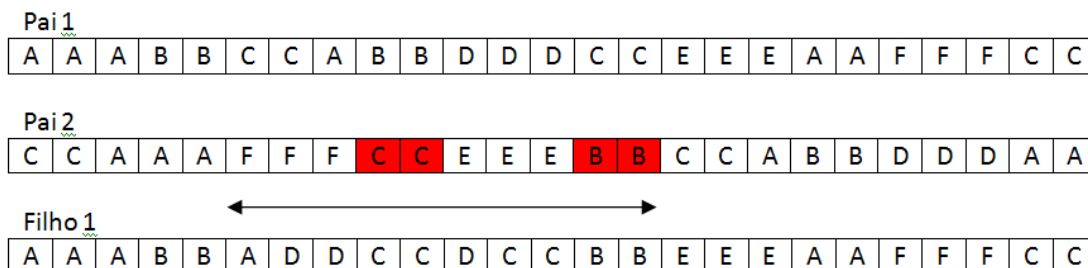


Figura 4.14 – Exemplo de cruzamento – Parte 6

Deve-se destacar também que, caso haja disciplinas cadastradas no sistema como obrigatórias em determinado horário, esses horários não serão envolvidos no cruzamento, sendo apenas copiados para os filhos.

Esse processo deve ser seguido da mesma forma para a geração do segundo filho, trocando-se apenas o lugar do primeiro e do segundo pai.

O cruzamento deve ser efetuado período por período para cada indivíduo.

4.4.6 – Mutação

Na operação de mutação, cada indivíduo gerado pelo processo de cruzamento poderá ter seu conteúdo alterado, caso seja satisfeita a condição da taxa de mutação escolhida pelo usuário na tela de parâmetros.

Durante o processo, são percorridos individualmente cada um dos períodos do indivíduo. Para cada período do indivíduo, serão selecionadas duas posições. Essas duas posições terão o seu conteúdo trocado.

A figura 4.15 a seguir ilustra esse processo, mostrando o conteúdo de apenas um período de um indivíduo após o processo de cruzamento. Foi realizado um sorteio das duas posições marcadas com a cor vermelha no primeiro filho. Essas posições terão o seu conteúdo trocado e o resultado é apresentado na segunda representação do respectivo filho.

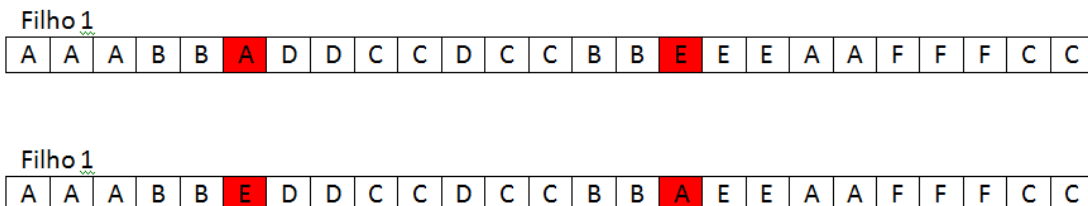


Figura 4.15 – Exemplo de mutação

É importante destacar que só ocorrerá troca de informações, caso os conteúdos envolvidos não estejam marcados como obrigatórios naquela posição. Caso isso aconteça, é escolhida outra posição.

4.4.7 – Condição de Parada

Quando dois pais são escolhidos no processo de seleção, eles passam pelos operadores de cruzamento e mutação para gerar dois filhos. Quando os dois filhos são gerados, são selecionados novamente dois pais para que possa ocorrer um novo cruzamento e uma nova mutação. Esse processo ocorre sucessivamente até que a quantidade de filhos gerados seja a mesma quantidade de indivíduos da população.

Quando isso ocorrer, o sistema utiliza o elitismo, que faz com que o melhor indivíduo encontrado em uma geração seja sempre mantido na próxima geração. Isso faz com que suas características genéticas sejam passadas adiante e sempre haja a possibilidade de que ele possa gerar descendentes ainda melhores nas próximas gerações.

Após o elitismo, a população antiga será desprezada e o Algoritmo Genético irá trabalhar com essa nova população criada. Nesse cenário, pode-se dizer que foi criada uma nova geração de grades horárias.

A condição de parada do sistema é a quantidade de gerações de grades horárias que serão criadas. Ou seja, o Algoritmo Genético será interrompido quando a quantidade de gerações percorridas durante a execução for igual à quantidade de gerações que foram determinadas nos parâmetros do sistema. A grade horária encontrada que, até esse momento, tiver o maior valor de aptidão será considerada a solução do problema.

4.5 – Geração da Grade Horária

A tela para geração de grade horária é acessada na tela principal do sistema, para isso basta clicar em Operações e depois em Gerar Grade. Essa tela é recomendada ao usuário final porque ela permite gerar e imprimir a grade horária de maneira rápida e não exige nenhum conhecimento sobre Algoritmos Genéticos.

Com a tela aberta, para se iniciar o processo, basta clicar no botão Gerar. A partir desse momento, pode-se acompanhar a evolução da geração das grades através de um indicador que mostra qual é a porcentagem já executada do processo. Após ser completado os 100%, a melhor grade horária obtida será mostrada na tela.

A figura 4.16 a seguir mostra a tela para geração de grades horárias antes de se iniciar o processo.

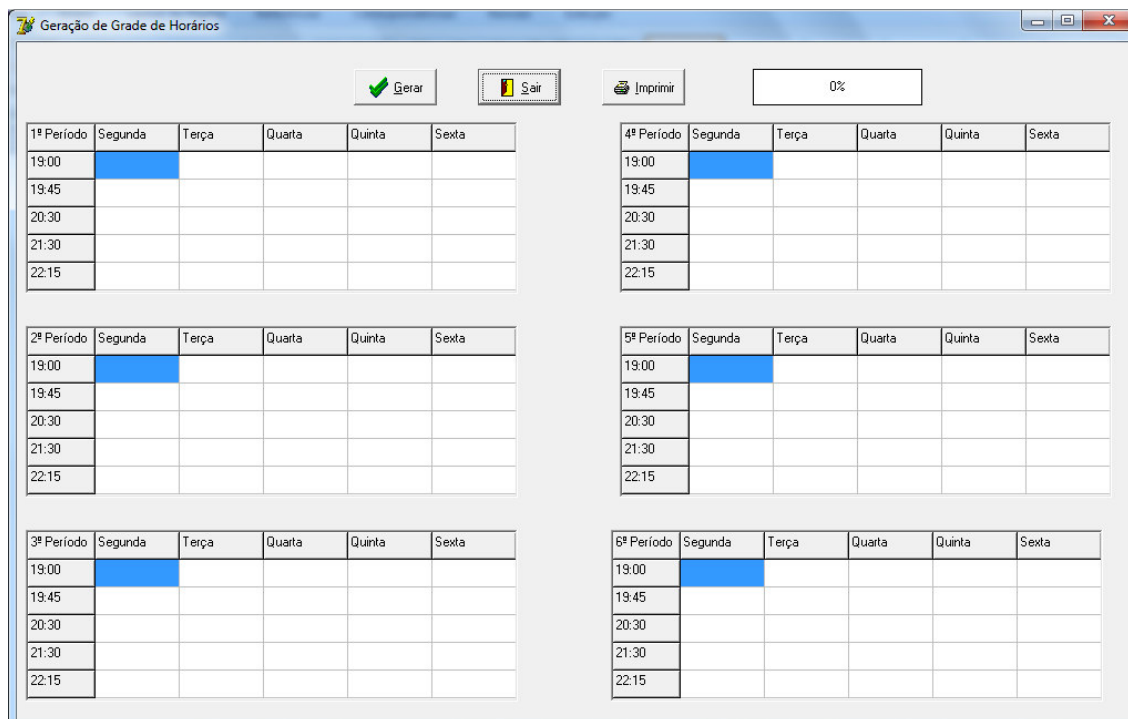


Figura 4.16 – Tela para geração da grade horária

Para verificar se a grade horária gerada atende a todas as restrições, basta verificar os horários que aparecem assinalados com cores diferentes. Os horários assinalados com a cor vermelha são os horários que violam a restrição que o professor não pode ministrar aula em um horário marcado como indisponível. Os horários assinalados com a cor verde são os horários que violam a restrição que um professor não pode ministrar duas aulas em diferentes turmas no mesmo horário. Essas duas restrições são restrições obrigatórias e não podem ser violadas. Uma grade horária apresentada como resultado final e que possua algum de seus horários marcado com essas cores, é uma grade horária impossível de ser implantada. Nesse caso é necessário gerar outra grade horária, clicando novamente no botão Gerar.

Caso após várias tentativas, não seja possível gerar uma grade horária que atenda a todas as restrições obrigatórias, pode ser necessário realizar uma alteração nos parâmetros do sistema. Nesse caso, é aconselhável que as alterações sejam executadas por um especialista em Algoritmos Genéticos, para que os parâmetros possam ser alterados e testados com o objetivo de se chegar a uma grade horária que atenda a todas as restrições obrigatórias.

Por último, os horários assinalados com a cor amarela são os horários que violam a restrição desejável que um professor não deve ministrar aulas em um horário assinalado como indesejável. Lembrando que, um horário indesejável é um horário no qual o professor pode ministrar aulas, porém não quer. Nesse caso, trata-se então de uma restrição que não torna a grade horária impossível de ser implantada, afinal o professor tem disponibilidade para ministrar aulas nesses horários. Para solucionar esse problema, basta o coordenador do curso conversar com o professor e informá-lo que

existe a necessidade que ele ministre aulas nesse horário para que a grade horária criada possa ser utilizada.

A figura 4.17 a seguir mostra uma tela na qual a grade horária já foi gerada e apresenta algumas restrições obrigatórias que ainda não foram satisfeitas no 2º, 3º, 5º e 6º períodos, ou seja, uma grade horária final que não representa uma solução do problema. No exemplo mostrado, o professor que ministra a disciplina do último horário da segunda-feira para o 2º e o 3º períodos é o mesmo. O professor que ministra a disciplina do segundo horário da terça-feira para o 5º e no 6º períodos também é o mesmo. Além disso, o professor que ministra a disciplina do segundo horário da quinta-feira para o 3º período não está disponível para ministrar aulas nesse horário. Existem ainda algumas restrições desejáveis que também não foram atendidas por essa grade. Para esse exemplo, é necessário executar novamente o processo para tentar encontrar uma nova grade horária melhor que a encontrada nessa execução, utilizando os mesmos parâmetros ou tentando alterá-los, por exemplo, aumentando a quantidade de gerações ou o número de indivíduos da população inicial. Dizer que uma grade horária gerada é melhor que outra, significa que ela possui menos restrições obrigatórias não atendidas e, em caso de empate, ela atende a mais restrições desejáveis. Afinal, uma grade horária dita como solução para o problema possui todas as restrições obrigatórias atendidas e o maior número de restrições desejáveis satisfeitas.

1º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	ALGOR -P	ALGOR -T	ALGOR -P	ATIV -P	PORT -T
19:45	PRO -T	ALGOR -T	PRO -T	ALGOR -T	ALGEB -T
20:30	ALGEB -T	PORT -T	ING -T	ALGEB -T	SI -T
21:30	PDI -P	PRO -T	ING -T	ALGOR -P	ATIV -P
22:15	ALGOR -P	SI -P	PDI -P	SI -P	ALGOR -T

2º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	ARQ -T	ESTR -P	EMPR -T	MOD -P	ESOF1 -T
19:45	ARQ -T	PROB -T	FM -T	FM -T	ESTR -T
20:30	ARQ -T	ESTR -T	ESOF1 -T	ARQ -T	FM -T
21:30	ESTR -P	ESTR -T	ESTR -P	MOD -T	MOD -P
22:15	FM -T	EMPR -T	PROB -T	ESOF1 -T	MOD -T

3º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	ESOF2 -T	PESQ -P	MD -T	PESQ -T	BD -P
19:45	POO -T	ESOF2 -T	MD -T	SO -P	SO -T
20:30	BD -T	PESQ -T	MD -T	POO -T	SD -P
21:30	ESOF2 -T	BD -P	PESQ -T	POO -P	PESQ -P
22:15	POO -P	BD -T	SO -T	BD -T	MD -T

4º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	RED -P	ADMDB -T	PV -P	ADMDB -T	PPI -P
19:45	RED -T	PV -P	MF -T	PPI -T	ADMDB -P
20:30	QUAL -T	PV -P	RED -T	PPI -P	PPI -T
21:30	QUAL -T	RED -P	MF -T	ADMDB -P	
22:15	QUAL -T	PV -P	RED -T	PV -P	PPI -T

5º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	DAW -T	ADE -T	SD -P	PDM -T	DAW -P
19:45	TADS -T	IMPL -T	ADE -T	SD -P	SD -T
20:30	PDM -P	PDM -T	TADS -T	DAW -P	TADS -T
21:30	IMPL -T	ADE -T	TADS -T	IMPL -P	SD -T
22:15	PDM -P	PDM -P	IMPL -P	DAW -T	

6º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	COM -T	MARK -T	SEG -P	NTC -T	COM -T
19:45	PDM -P	PDM -T	GPS -T	PDM -P	NTC -T
20:30	SEG -P	GPS -T	GPS -T	GPS -T	PDM -T
21:30	SEG -T	MARK -T	NTC -T	MARK -T	COM -T
22:15	SEG -T	SEG -T	COM -T	PDM -T	NTC -T

Figura 4.17 – Grade horária gerada que não atende todas as restrições obrigatórias

Pode-se então dizer que, uma grade gerada que não possua horários assinalados com as cores vermelha e verde, é uma grade que atende a todas as restrições obrigatórias e pode ser considerada como uma solução final do problema.

Se a grade horária final apresentar horários assinalados apenas com a cor amarela, caso o usuário queira, ele pode tentar gerar novamente outra grade com os mesmos parâmetros ou tentar alterar os parâmetros, para tentar eliminar também as restrições desejáveis.

A figura 4.18 a seguir mostra uma nova execução do Algoritmo Genético, na qual, todas as restrições obrigatórias foram atendidas. Embora ainda existam algumas restrições desejáveis ainda não atendidas, essa grade horária representa uma solução viável para o problema e pode ser implantada na instituição.

1º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	ALGEB -T	ALGOR -T	ING -T	ATIV -P	ALGOR -T
19:45	ALGEB -T	ALGOR -T	ING -T	ALGOR -P	ALGOR -T
20:30	ALGEB -T	PRO -T	PDI -P	ALGOR -P	SI -P
21:30	PRO -T	PDI -P	ALGOR -P	PORT -T	ATIV -P
22:15	PRO -T	PORT -T	ALGOR -P	SI -T	SI -P

2º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	ARQ -T	MOD -P	FM -T	ESTR -T	ESOF1 -T
19:45	ESTR -P	MOD -T	PROB -T	ESTR -T	ESOF1 -T
20:30	ESTR -P	MOD -P	EMPR -T	ESTR -T	PROB -T
21:30	ARQ -T	FM -T	ARQ -T	ARQ -T	MOD -T
22:15	FM -T	EMPR -T	ESTR -P	ESOF1 -T	FM -T

3º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	BD -P	ESOF2 -T	MD -T	PESQ -T	P00 -T
19:45	BD -P	ESOF2 -T	MD -T	PESQ -T	SO -P
20:30	ESOF2 -T	BD -T	P00 -T	PESQ -T	SO -T
21:30	P00 -P	PESQ -P	SO -P	BD -T	MD -T
22:15	P00 -P	PESQ -P	SO -T	BD -T	MD -T

4º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	RED -T	RED -P	PV -P	ADMBD -P	PPI -T
19:45	QUAL -T	ADMBD -P	QUAL -T	ADMBD -T	PPI -P
20:30	ADMBD -T	PV -P	QUAL -T	MF -T	PPI -T
21:30	MF -T	PV -P	RED -P	PPI -P	PV -P
22:15	RED -T	PV -P	RED -T	PPI -T	

5º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	DAW -T	PDM -T	ADE -T	TADS -T	IMPL -T
19:45	PDM -P	PDM -T	TADS -T	IMPL -T	DAW -T
20:30	PDM -P	ADE -T	PDM -P	IMPL -P	
21:30	TADS -T	ADE -T	SD -P	DAW -P	SD -T
22:15	TADS -T	IMPL -P	SD -P	DAW -P	SD -T

6º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	COM -T	MARK -T	SEG -T	NTC -T	PDM -T
19:45	SEG -T	MARK -T	SEG -P	NTC -T	PDM -T
20:30	SEG -T	COM -T	SEG -P	NTC -T	GPS -T
21:30	GPS -T	PDM -P	MARK -T	PDM -T	COM -T
22:15	GPS -T	GPS -T	NTC -T	PDM -P	COM -T

Figura 4.18 – Grade horária gerada que atende todas as restrições obrigatórias

O sistema também permite a geração de grade horária no sistema através de uma tela de ajuste de parâmetros. Para acessá-la, basta clicar em Operações e depois em Ajustar Parâmetros. A diferença é que nessa tela, além de ser possível ajustar os parâmetros de execução do Algoritmo Genético, é possível realizar alguns testes para verificar se os parâmetros estão corretamente ajustados. Esses testes são realizados através de alguns indicadores gráficos que tornam a execução do Algoritmo Genético consideravelmente mais lenta. Sendo assim, essa tela é recomendada apenas para ajustes e testes dos parâmetros por um especialista em Algoritmos Genéticos e, nunca para execução corriqueira do sistema por um usuário comum.

Os indicadores gráficos foram criados para ajudar o especialista em Algoritmos Genéticos a encontrar os melhores parâmetros, que se aplicam a execução das operações para aquele conjunto de informações cadastradas.

Existem dois indicadores com esse objetivo. O primeiro deles consiste em mostrar no decorrer das gerações, qual é a grade que, até aquela geração, melhor atende ao problema. Ou seja, na

medida em que as gerações vão sendo percorridas, a melhor grade até então encontrada será apresentada na tela. A melhor grade é a grade que possui menos restrições obrigatórias não atendidas. Essa melhor grade será mostrada com seus horários assinalados, com as cores vermelha, verde e amarela, quando existirem restrições ainda não atendidas, conforme descrito anteriormente.

A figura 4.19 a seguir mostra uma grade horária intermediária gerada pelo sistema, com várias restrições obrigatórias e desejáveis não atendidas.

1º Período	Segunda	Terça	Quarta	Quinta	Sexta	4º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	PRO -T	ALGOR -T	ING -T	ALGOR -P	ALGOR -P	19:00	ADMBD -T	PV -P	RED -T	PPI -P	ADMBD -T
19:45	ALGEB -T	ALGOR -T	ALGOR -P	SI -P	ATIV -P	19:45	RED -P	PV -P	QUAL -T	MF -T	PPI -P
20:30	PRO -T	PRO -T	ALGOR -T	ALGOR -T	PORT -T	20:30	PV -P	QUAL -T	PV -P	PPI -T	MF -T
21:30	ALGOR -P	PDI -P	ING -T	ALGEB -T	SI -T	21:30	RED -T	ADMBD -P	QUAL -T	PPI -T	
22:15	PDI -P	PORT -T	ALGEB -T	ATIV -P	SI -P	22:15	PV -P	RED -T	RED -P	ADMBD -P	PPI -T
2º Período	Segunda	Terça	Quarta	Quinta	Sexta	5º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	ARQ -T	MOD -T	ESTR -T	ESOF1 -T	PROB -T	19:00	DAW -P	IMPL -P	SD -T	TADS -T	DAW -T
19:45	ESTR -P	EMPR -T	ARQ -T	FM -T	PROB -T	19:45	TADS -T	AOE -T	PDM -P	DAW -T	PDM -P
20:30	MOD -T	FM -T	ESOF1 -T	FM -T	ESTR -T	20:30	PDM -T	PDM -T	PDM -P	DAW -P	SD -P
21:30	FM -T	EMPR -T	MOD -P	MOD -P	ESTR -T	21:30	IMPL -T	IMPL -P	AOE -T	TADS -T	SD -T
22:15	ARQ -T	ESTR -P	ARQ -T	ESTR -P	ESOF1 -T	22:15	TADS -T	IMPL -T	AOE -T	SD -P	
3º Período	Segunda	Terça	Quarta	Quinta	Sexta	6º Período	Segunda	Terça	Quarta	Quinta	Sexta
19:00	SO -P	BD -P	MD -T	POO -P	PESQ -P	19:00	SEG -T	MARK -T	PDM -T	NTC -T	COM -T
19:45	ESOF2 -T	ESOF2 -T	SO -P	SO -T	POO -T	19:45	PDM -T	SEG -P	NTC -T	NTC -T	GPS -T
20:30	BD -T	BD -T	MD -T	BD -T	MD -T	20:30	SEG -T	GPS -T	SEG -P	PDM -T	PDM -P
21:30	BD -P	PESQ -T	SO -T	POO -P	PESQ -T	21:30	MARK -T	PDM -P	COM -T	GPS -T	NTC -T
22:15	ESOF2 -T	PESQ -P	MD -T	PESQ -T	POO -T	22:15	SEG -T	MARK -T	COM -T	GPS -T	COM -T

Figura 4.19 – Grade horária intermediária gerada no sistema

O segundo indicador consiste em dois gráficos, mostrados no canto direito da tela de ajuste de parâmetros, que são atualizados a cada geração percorrida pelo Algoritmo Genético. Eles representam o valor da aptidão da melhor grade horária encontrada até o momento e o valor médio das aptidões das grades horárias da geração atual. Em ambos, o eixo X representa a quantidade de gerações já percorridas e o eixo Y, representa o valor da aptidão.

A figura 4.20 a seguir mostra um exemplo dos dois gráficos gerados para 100 gerações percorridas.

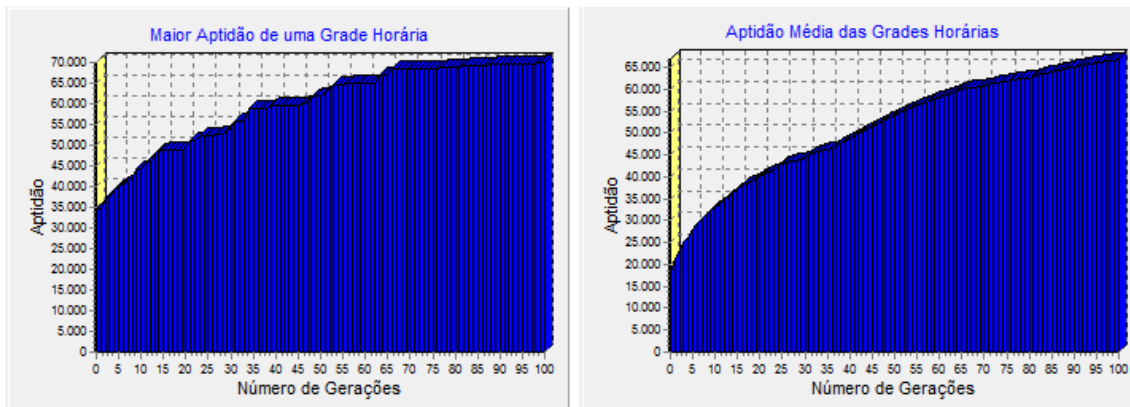


Figura 4.20 – Gráficos da maior aptidão encontrada e da média

Graças a esses dois indicadores gráficos, é possível comprovar que o Algoritmo Genético consegue encontrar a cada geração percorrida melhores grades horárias. Afinal, pode-se observar que na medida em que são percorridas as gerações, a grade horária que está sendo mostrada na tela, que é a atual melhor grade encontrada, possui cada vez menos restrições não atendidas. Em outras palavras, pode-se observar que a cada geração são encontradas na melhor grade, cada vez menos restrições, que são representadas pelos horários assinalados com cores vermelha, verde e amarela.

Através dos gráficos, pode-se observar que, tanto o valor da aptidão da melhor grade encontrada até o momento, quanto o valor médio das grades horárias da geração atual, cresce a cada geração percorrida.

Esses fatos comprovam que, através das operações realizadas pelo Algoritmo Genético, são encontradas melhores grades horárias a cada geração, pois as características existentes nos indivíduos de uma geração que atendem a solução do problema são propagadas para a próxima geração. Esse processo faz com que possa se encontrar em determinada geração uma grade horária que atenda todas as restrições obrigatórias desejadas pela instituição. Sendo assim pode-se dizer que, com os parâmetros corretamente ajustados, as grades horárias evoluem no decorrer das gerações, seguindo os conceitos da teoria da evolução das espécies.

4.6 – Geração dos Relatórios

O módulo de geração de relatórios é o responsável por permitir ao usuário do sistema realizar a impressão das informações geradas pelo módulo de geração da grade horária.

Após a geração da grade, o sistema permite a impressão de dois tipos de relatórios: o relatório completo com a grade horária semanal do curso e o relatório com as aulas que cada professor irá ministrar durante a semana. Para visualizar essa opção, basta clicar no botão Imprimir, quando a grade já tiver sido gerada e estiver mostrada na tela. Nessa situação, será aberta uma janela, onde o usuário deverá optar por um dos relatórios.

A figura 4.21 a seguir mostra a tela onde o usuário seleciona qual relatório será gerado.

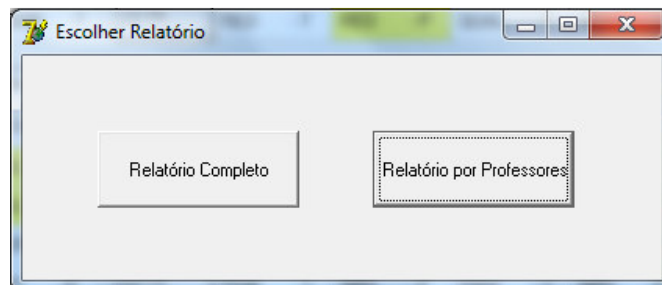


Figura 4.21 – Tela de seleção de relatório

As particularidades dos dois tipos de relatórios serão apresentadas no ANEXO A.

5 – RESULTADOS E DISCUSSÃO

5.1 – Estrutura do Curso

O curso de Análise e Desenvolvimento de Sistemas do IFTM, campus Uberaba é realizado em período noturno, com aulas ministradas de segunda a sexta-feira, no horário entre 19:00 e 23:00. São 5 aulas diárias, totalizando 25 aulas semanais.

O curso tem duração de 3 anos e está dividido em 6 períodos.

O quadro 5.1 a seguir mostra as disciplinas oferecidas durante o curso, em qual período são oferecidas e qual é a sua quantidade de aulas semanais.

Período	Disciplina	Quantidade de Aulas
1º	Algoritmos e Lógica de Programação	8
1º	Planejamento de Projetos	3
1º	Sistemas de Informação	3
1º	Processamento de Imagens	2
1º	Inglês	2
1º	Álgebra	3
1º	Atividade Complementar	2
1º	Português	2
2º	Estrutura de Dados	6
2º	Fundamentos Matemáticos	4
2º	Modelagem de Banco de Dados	4
2º	Empreendedorismo	2
2º	Arquitetura de Organização de Computadores	4
2º	Engenharia de Software 1	3
2º	Probabilidade e Estatística	2
3º	Banco de Dados	5
3º	Programação Orientada a Objetos	4
3º	Engenharia de Software 2	3
3º	Matemática Discreta	4
3º	Sistemas Operacionais	4
3º	Pesquisa e Ordenação	5
4º	Programação para Internet	5
4º	Administração de Banco de Dados	4
4º	Redes de Computadores	5
4º	Programação Visual	5

4º	Qualidade de Software	3
4º	Matemática Financeira	2
5º	Tópicos Avançados de Desenvolvimento de Softwares	4
5º	Programação de Dispositivos Móveis	5
5º	Desenvolvimento de Aplicações Web	4
5º	Administração e Organização de Empresas	3
5º	Implantação de Servidores	4
5º	Sistemas Distribuídos	4
6º	Segurança de Redes	5
6º	Gerência e Projeto de Softwares	4
6º	Comércio Eletrônico	4
6º	Marketing	3
6º	Legislação	2
6º	Ética	2
6º	Inteligência Artificial	4

Quadro 5.1 – Disciplinas do curso de Análise e Desenvolvimento de Sistemas

O corpo docente do curso é composto de 18 professores, sendo eles graduados, especialistas, mestres ou doutores, alguns com sua titulação concluída, outros ainda ampliando seu nível de titulação. O quadro 5.2 a seguir apresenta o corpo docente do curso e a titulação de cada um dos seus integrantes.

Professor	Titulação
Adolfo Modesto Gil	Especialista
Alexandre Ribeiro da Silva	Mestrando
Antenor Roberto Pedroso	Mestre
Camilo Lélis Tosta de Paula	Mestrando
Cassimiro Rodrigues Filho	Graduado
Clidenor Ferreira de Araújo Filho	Mestre
Daniela Resende Silva Orbolato	Mestre
Ernani Viriato de Melo	Mestre
Gustavo Marino Botta	Especialista
Hugo Leonardo Pereira Rufino	Doutor
Jefferson Beethoven Martins	Mestrando
Johann Max Hofmann	Mestre
José Ricardo Gonçalves Manzan	Mestrando
Ludmila Martins Costa	Graduada

Marcelo da Silva Barreiro	Mestre
Marcelo Sedassari Galvão	Graduado
Marcelo Ponciano da Silva	Doutorando
Paula Nakamoto	Doutora
Rafael Godoi Orbolato	Mestre
Sandra Chaves Gardellari	Doutora
Vicente Batista dos Santos Neto	Mestre
Wagner Jacinto de Oliveira	Mestre

Quadro 5.2 – Corpo docente do curso de Análise e Desenvolvimento de Sistemas

5.2 – Processo de geração da grade horária

Antes do desenvolvimento do sistema apresentado nesse trabalho, a geração da grade horária do curso de Análise e Desenvolvimento de Sistemas do IFTM, era realizada manualmente. O coordenador de curso, após a divisão das disciplinas entre os professores, conversava com cada um deles para descobrir a sua disponibilidade e a partir daí, gastava dias para montar um horário que satisfizesse todas as restrições.

Um transtorno maior ocorria caso, após o horário estar montado, surgisse alguma alteração, como por exemplo, um professor ter sua disponibilidade alterada por algum motivo. Nessa situação, o horário deveria ser novamente alterado e o coordenador de curso gastaria novamente muito tempo para remontar a grade horária.

Com a utilização do sistema, o coordenador de curso precisa apenas realizar os cadastros necessários, de cursos, períodos, professores e disciplinas. Com os cadastros efetuados, o sistema pode gerar, em poucos minutos, uma grade horária que atenda a todas as restrições.

Caso após a grade horária estar montada, surja alguma alteração na disponibilidade de algum professor ou em alguma restrição cadastrada no sistema, basta alterar o cadastro necessário e gerar uma nova grade horária em poucos minutos.

No futuro, quando o sistema for montar a grade horária para outro semestre, a tarefa se torna ainda mais simples, pois somente será necessário alterar as informações que tiverem sido modificadas na grade horária, como um cadastro de um novo professor, alteração de alguma disciplina da grade curricular ou a alteração do professor que ministra uma determinada disciplina.

Para realizar o cadastro dos professores e das disciplinas, foi encaminhada pelo coordenador do curso para cada um dos professores, uma ficha para preenchimento de algumas informações como:

- Disponibilidade para ministrar aulas, descrita como disponível, indisponível ou indesejado para cada horário de cada um dos dias da semana;
- Disciplinas que irá ministrar;
- Divisão das aulas da disciplina entre teóricas e práticas;

- Divisão das aulas teóricas e práticas, indicando a preferência em ministrar aulas em horários simples, duplos ou triplos;
- Se as disciplinas possuem horários obrigatórios e, caso possuam, quais são esses horários.

A ficha que foi encaminhada aos professores está no ANEXO B desse trabalho.

5.3 – Análise dos Resultados

Após a realização do cadastro das informações reais que foram recolhidas através das fichas preenchidas pelos professores, o próximo passo do processo é o ajuste dos parâmetros. Para se obter um melhor desempenho do Algoritmo Genético, é necessário encontrar o conjunto de parâmetros que melhor se aplica ao conjunto de dados cadastrados. Para encontrá-los, foram realizados diversos testes no sistema, com diversos valores de parâmetros. Alguns dos resultados obtidos foram considerados satisfatórios e outros, não. Com base nos valores satisfatórios, foi possível refinar o conjunto de parâmetros para se encontrar resultados ainda mais satisfatórios, em relação ao tempo gasto para que o AG encontre uma grade horária que atenda todas as restrições obrigatórias e em relação ao número de gerações percorridas até que isso ocorra. Através dos testes que produziram valores não satisfatórios, foi possível determinar quais ajustes de valores de parâmetros não contribuam para uma melhoria nos resultados obtidos.

Para a realização dos testes, foi utilizado um notebook com processador Turion X2 de 2,30 GHz, 4 GB de memória RAM, trabalhando com o Windows 7 Professional como sistema operacional.

Os testes foram realizados para verificar qual era o valor recomendado para a população inicial, a taxa de cruzamento, a taxa de mutação, o número de gerações a serem percorridas, o tipo de cruzamento efetuado e o número de pontos do cruzamento para esse conjunto de informações cadastradas.

Para realização dos testes, é utilizada a tela para ajuste de parâmetros, pois ela apresenta os indicadores gráficos que são necessários para a avaliação dos resultados obtidos pelo Algoritmo Genético.

O quadro 5.3 a seguir mostra os testes efetuados, a coluna Gerações mostra a média da quantidade de gerações necessárias para eliminar todas as restrições obrigatórias. A coluna Tempo mostra a média do tempo gasto para que o sistema consiga eliminar todas as restrições obrigatórias. Para medir esse tempo, foi utilizada a tela de alteração de parâmetros. Ou seja, o tempo gasto para que o sistema encontre um resultado que atenda a todas as restrições obrigatórias é ainda menor, pois a geração das grades horárias será realizada pelo usuário final na tela de geração da grade, que é consideravelmente mais rápida. Sendo assim, com o melhor conjunto de parâmetros testado, é possível gerar uma grade horária válida em pouco mais de dois minutos. Foram realizados testes com cruzamento utilizando 1 ou 2 pontos de cruzamento, utilizando como método de seleção de pais, o torneio e a roleta. Foram testados diversos valores de população inicial, taxa de cruzamento e taxa de

mutação. Os valores de todos os parâmetros foram sendo alterados de acordo com os resultados que foram sendo obtidos, com a intenção de encontrar uma solução que atenda ao problema em menos tempo e percorrendo um número menor de gerações.

Cruzamento	Pts. De Cruz.	Pop. Inicial	Tx. Cruzamento	Tx. Mutação	Gerações	Tempo
Torneio	1	1000	0,9	0,01	54	02:25
Torneio	2	1000	0,9	0,01	71	02:53
Torneio	1	2000	0,9	0,01	59	04:01
Torneio	2	2000	0,9	0,01	98	05:36
Torneio	1	5000	0,9	0,01	64	08:04
Torneio	2	5000	0,9	0,01	105	13:15
Torneio	1	1000	0,9	0,03	57	02:36
Torneio	2	1000	0,9	0,03	84	03:16
Torneio	1	2000	0,9	0,03	62	04:25
Torneio	2	2000	0,9	0,03	91	04:41
Torneio	1	5000	0,9	0,03	116	13:37
Torneio	2	5000	0,9	0,03	107	13:15
Torneio	1	1000	0,95	0,01	58	02:37
Torneio	2	1000	0,95	0,01	82	02:59
Torneio	1	2000	0,95	0,01	59	04:03
Torneio	2	2000	0,95	0,01	90	05:55
Torneio	1	1000	0,9	0,005	65	02:59
Torneio	2	1000	0,9	0,005	85	03:22
Torneio	1	2000	0,9	0,005	68	04:25
Torneio	2	2000	0,9	0,005	99	06:12
Roleta	1	1000	0,9	0,01	2650	01:08:00
Roleta	1	1000	0,9	0,01	3540	01:32:20
Roleta	2	2000	0,9	0,01	2843	01:15:30
Roleta	2	2000	0,9	0,01	4022	01:39:43

Quadro 5.3 – Testes efetuados para encontrar os parâmetros do sistema

Quando se analisa os valores encontrados nos testes, se nota que quanto maior a população inicial, mais tempo é gasto para que o sistema encontre uma grade horária que atenda a todas as restrições obrigatórias. Foram utilizados, valores de população inicial de 1000, 2000 e 5000 indivíduos para testes em torneio e, utilizando 1000 indivíduos, foi possível eliminar todas as restrições obrigatórias em um tempo menor. Quando é utilizada uma população inicial inferior a 1000 indivíduos, na maioria dos testes efetuados, o sistema não consegue encontrar uma solução viável.

Os valores da taxa de mutação e da taxa de cruzamento definidos foram de 0,01 e 0,9 respectivamente. Para se chegar a esses valores, foi verificado que, aumentando ou reduzindo a taxa de mutação ou a taxa de cruzamento, tanto o tempo gasto, quanto a quantidade de gerações percorridas até que todas as restrições obrigatórias fossem atendidas, aumenta.

Foram realizados testes com o cruzamento sendo efetuado em um ou dois pontos e, em todos eles, o cruzamento em apenas um ponto se mostrou mais eficaz.

Sobre o método de seleção dos pais para a realização da operação de cruzamento, foram realizados testes utilizando a roleta e o torneio. Foi verificado que, utilizando a roleta, não se consegue encontrar uma grade horária que satisfaça as restrições em tempo hábil, mesmo variando todos os parâmetros. Por isso, foi constatado que, para o conjunto de dados cadastrados, a utilização do torneio é mais recomendada.

É importante destacar que os parâmetros recomendados variam de acordo com os dados que são manipulados pelo Algoritmo Genético. Ou seja, caso a base de dados seja alterada, através de alteração no corpo docente do curso, na disponibilidade dos professores, dentre outros, devem ser realizados novos testes na tela de parâmetros para determinar o novo conjunto de parâmetros recomendado. Esse conjunto de parâmetros deve ser capaz de gerar uma grade horária que atenda a todas as restrições obrigatórias, tente também atender as restrições desejáveis e não gaste um tempo considerável para isso.

A questão do tempo gasto para encontrar o resultado foi uma das maiores preocupações durante o todo o processo de desenvolvimento e teste do *software*. Afinal, não era viável que o sistema gerasse uma grade horária que atendesse ao problema, mas que demorasse muito tempo para concretizar esse processo. Inicialmente, o *software* não encontrava um resultado satisfatório porque demorava muito para percorrer cada geração. O processo para geração da grade horária chegava a durar várias horas. Sendo assim, se tornava inviável aguardar até que o sistema conseguisse atender todas as restrições e retornasse uma grade horária que pudesse ser implantada e que não violasse nenhuma das condições impostas.

Após a realização de vários testes, foi constatado que a demora ocorria durante o cálculo do valor da aptidão do indivíduo por dois motivos.

O primeiro motivo era que, para cada horário de aula de cada período de cada uma das grades horárias, era realizada uma consulta na base de dados. Essa consulta tinha como objetivo verificar qual era a disponibilidade cadastrada para o professor responsável pela disciplina alocada para aquele horário e, com base nessa disponibilidade, atribuir um valor de bônus ou penalidade a grade horária analisada. Como uma grade horária de 6 períodos possui 150 horários, eram realizadas 150 consultas a base de dados para cada grade horária. Analisando uma população de 1000 indivíduos, seriam realizadas 150000 consultas para cada geração.

O segundo fator que fazia com que o tempo gasto para o cálculo da função de aptidão fosse alto era que, para cada indivíduo, era montada através de sucessivas consultas na base de dados, uma grade paralela que possui como conteúdo o código de todos os professores que estão ministrando cada uma das aulas dos períodos. O objetivo da criação dessa grade de professores era realizar uma comparação entre os períodos de cada grade horária para verificar se um professor estava ministrando aula em dois lugares ao mesmo tempo e penalizar a grade horária nessa situação. Como para cada horário da grade era realizada uma consulta a base de dados, chegava-se como no parágrafo anterior a

mais 150 consultas a base de dados para cada grade horária gerada e 150000 consultas para cada geração, se for considerada uma população de 1000 indivíduos.

Esses dois fatores citados faziam com que houvesse um excesso de consultas na base de dados causando uma lentidão inaceitável no sistema.

A partir do momento em que foram diagnosticadas quais eram as causas da lentidão do sistema, foram realizadas alterações para diminuir a quantidade de consultas as bases de dados.

Primeiramente, a grade paralela de professores que, é utilizada para verificação de conflito de um professor ministrando aula em dois lugares ao mesmo tempo, passou a ser gerada no mesmo instante em que era verificada a disponibilidade do professor para cada horário de cada período. Fato esse que já melhorou muito o desempenho do sistema, pois as consultas a base de dados foram reduzidas a metade.

Em um segundo momento, a verificação da disponibilidade do professor para cada horário de cada período de cada indivíduo, foi substituída por uma verificação por disciplina. Nessa nova verificação, cada disciplina é buscada em cada uma das posições em todos os indivíduos da população. Quando localizada, é verificada a disponibilidade do professor para aquele horário e atribuído um valor de bônus ou penalidade ao indivíduo. Utilizando esse método, é realizada uma consulta na base para cada disciplina cadastrada no sistema, enquanto no método anterior era realizada uma consulta para cada horário de cada período de cada indivíduo. Ou seja, a diminuição do número de consultas a base de dados foi considerável.

A última melhoria realizada no sistema foi o desenvolvimento da tela para geração da grade horária sem as opções gráficas de visualização. Embora essas visualizações gráficas sejam muito eficazes para avaliar se e quando o Algoritmo Genético está encontrando a solução que satisfaz ao problema em tempo hábil, elas são atualizadas a cada geração, no caso dos gráficos de aptidão máxima encontrada e aptidão média da geração atual, e a cada alteração da melhor grade encontrada, no caso das grades mostradas com horários assinalados em cores diferentes na tela. Essas atualizações consomem um tempo considerável e não são necessárias para o usuário final, pois a ele interessa apenas o resultado final encontrado que satisfaça o problema em menor tempo possível.

6 – CONCLUSÃO E TRABALHOS FUTUROS

O trabalho propõe um sistema utilizando algoritmos genéticos que seja capaz de solucionar um problema com restrições do mundo real. O estudo de caso apresenta a construção de um sistema gerador de grade horária que atenda a todas as restrições obrigatórias e que consiga atender também as restrições desejáveis do curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro – campus Uberaba.

Para isso, foi utilizada a técnica de Algoritmos Genéticos que se mostrou eficiente para esta situação. Afinal, através dela foi possível solucionar um problema do mundo real, automatizar um processo no qual o coordenador de curso gastava horas e até mesmo dias, para gerar manualmente uma grade horária que atendesse as restrições impostas. O sistema permite gerar uma grade horária que atenda todas as restrições obrigatórias em poucos minutos.

O sistema implementado é flexível, pois permite que sejam facilmente inseridas novas restrições ou alteradas as restrições já existentes, bastando para isso inserir as modificações necessárias através das janelas de interface correspondentes ou, ainda, executar uma alteração na função de aptidão do Algoritmo Genético.

O sistema desenvolvido também pode ser utilizado para fins didáticos, pois através dele é possível observar o funcionamento do Algoritmo Genético, através dos indicadores gráficos que mostram a evolução das grades horárias no decorrer das gerações. É possível também permitir ao estudante testar diversos parâmetros do AG, para que se possa verificar qual a influência desse ajuste no resultado final.

Além dessas contribuições, pode-se destacar o desenvolvimento da estratégia de cruzamento que impede que sejam gerados filhos que sejam incompatíveis com a realidade da grade horária do curso, ou seja, que possuam uma quantidade de aulas de cada disciplina diferente da que está cadastrada no sistema.

Pode-se também destacar as melhorias implantadas no sistema em relação a diminuição da quantidade de consultas realizadas a base de dados e, conseqüentemente, a diminuição no tempo gasto para a geração da grade horária.

Como trabalhos futuros, pode-se estender a criação de grades horárias para todos os cursos da área de informática do IFTM ou até mesmo para cursos de outras áreas.

Outro avanço muito significativo pode ser alcançado, caso o sistema seja ampliado para também atender outra necessidade da instituição: a alocação dos laboratórios e das salas utilizadas para as aulas.

Em relação à codificação do sistema, outro avanço possível de ser alcançado no trabalho consiste em melhorar o processamento da aplicação, utilizando-se processamento paralelo. O processamento paralelo será essencial caso se amplie o sistema, para que ele gere a grade horária para

outros cursos do IFTM ou mesmo trabalhe com outros tipos de variáveis como alocação de espaço. Caso contrário, o tempo gasto para geração de uma grade horária que atenda a todas as restrições será muito alto e inviável.

Outra otimização que pode ser implantada é o ajuste automático do cromossomo que representa a grade horária. Na situação atual, a estrutura armazena o conteúdo de seis períodos, pois representa a realidade do curso de Análise e Desenvolvimento de Sistemas. Para que seja possível processar a grade horária de outros cursos, é aconselhável que o cromossomo seja ajustado automaticamente de acordo com a quantidade de períodos do curso com o qual se planeja trabalhar.

Outra possibilidade consiste em investigar outras maneiras mais eficientes de armazenar o cromossomo que representa a grade horária. A atual estrutura consiste em uma matriz bidimensional na qual são armazenados elementos alfanuméricos que representam as disciplinas do curso. Podem ser conduzidos estudos com o intuito de descobrir outras possibilidades que podem acarretar um menor processamento computacional e, como consequência, diminuir o tempo gasto para geração da grade horária.

Pode-se dizer que o resultado alcançado pelo trabalho foi altamente satisfatório, visto que no segundo semestre do ano de 2012, a grade horária utilizada pelo curso de Análise e Desenvolvimento de Sistemas foi gerada através do sistema desenvolvido. Para os próximos semestres, o processo será ainda mais simples, visto que boa parte dos dados cadastrados será mantida, pode ser necessário realizar apenas algumas alterações, como por exemplo, a disponibilidade dos professores, o professor responsável por alguma disciplina, alguma mudança a grade curricular do curso, dentre outras possibilidades.

REFERÊNCIAS:

BITTENCOURT, G. Inteligência artificial: ferramentas e teorias. Florianópolis: Editora da UFSC, 2006.

BORSOI, M. H. G. et al. Sistema de criação e gerenciamento de grade horária utilizando o método Monte Carlo. In: XI Encontro Latino Americano de Iniciação Científica, 2007.

CAMPOS, F. N. Algoritmo genético na resolução do problema da grade horária. Trabalho de Conclusão de Curso. Salvador: Universidade Federal da Bahia, 2002.

CARVALHO, A. P. L. F. Algoritmos Genéticos. Disponível em: <<http://www.icmc.usp.br/~andre/research/genetic>>. Acesso em: 04 abr. 2012.

CORMEN, T. H. et al. *Algoritmos – Teoria e Prática*. 2ª edição. Rio de Janeiro: Editora Campus, 2002.

FRANCISCO, D. O.; SILVA, I. N. Estratégia de otimização baseada em sistemas inteligentes aplicada ao problema de geração de grades horárias. In: 10th Brazilian Congress on Computational Intelligence – CBIC, 2011.

FREITAS, C. C. et al. Uma ferramenta baseada em algoritmos genéticos para a geração de tabela de horário escolar. In: Sétima Escola Regional de Computação Bahia-Sergipe, 2007.

FUCILINI, T. A. P.; LODI, M. L.; REBONATTO, M. T. Timetabling com algoritmos genéticos paralelos. In: 9ª Escola Regional de Alto Desempenho (ERAD 2009).

HEINEN, M. R.; OSÓRIO, F. S. Algoritmos genéticos aplicados ao problema de roteamento de veículos, Revista Hífen, Uruguaiana, V. 30, N.58, 2006.

IFTM. Página do curso de Análise e Desenvolvimento de Sistemas. Disponível em: <<http://www.iftm.edu.br/uberaba/cursos/superiores/analise/index.php>>. Acesso em 26 jul. 2012.

JÚNIOR, A. F. L. J.; ROCHA, C. A. J. AGHORA: algoritmos genéticos para geração de horários de aula. In: XIX Semana Paraense de Informática e Telecomunicações, II Congresso de Tecnologia de Informação e Comunicação da Amazônia, 2005.

LINDEN, R. Algoritmos Genéticos. 3ª Edição. Rio de Janeiro: Editora Ciência Moderna Ltda., 2012.

LOBO, E. L. M. Uma solução do problema de horário escolar via algoritmo genético paralelo. Dissertação de Mestrado. Belo Horizonte: CEFET/MG, 2005.

PERANCONI, D. S.; NOTHAFT, O. E.; SCHREIBER, J. N. Sistema gerador de horários utilizando algoritmos genéticos paralelos. In: II Salão de Ensino e de Extensão. Universidade de Santa Cruz do Sul, 2011.

PREIS, T. A. Protótipo gerador de grades horárias para instituições de ensino. Trabalho de Conclusão de Curso. Blumenau: Universidade Regional de Blumenau, 2007.

REZENDE, S. O. Sistemas inteligentes: fundamentos e aplicações. Barueri/SP: Manole, 2005.

RICH, E. Inteligência Artificial. 2ª Edição. São Paulo: Makron Books, 1993.

SANTOS, C. E. S. Algoritmo genético empregado na alocação de estados em máquinas markovianas. Dissertação de Mestrado. Ilha Solteira: Universidade Estadual Paulista, 2005.

SCHAERF, A. A Survey of Automated Timetabling. In: CENTRUM voor Wiskunde en Informatica (CWI). Report CS-R9567. Amsterdam, 1995.

TAURION, C. Cloud Computing: computação em nuvem: transformando o mundo da tecnologia da informação. Rio de Janeiro: Brasport, 2009.

TIMES'COOL. O problema da formação de horários. Disponível em: <<http://timescool.lotimiza.com/>>. Acesso em 09 abr. 2012.

URANIA. Programa para fazer horário escolar. Disponível em: <<http://www.horario.com.br/programa/>>. Acesso em 09 de abr. 2012.

VIEIRA, F.; MACEDO, H. Sistema de alocação de horários de cursos universitários: um estudo de caso no departamento de computação da Universidade Federal de Sergipe, Revista Scientia Plena, São Cristóvão/SE, V.7, N.3, 2011.

WANG, Z., LIU, J. L., YU, X. Self-Fertilization Based Genetic Algorithm for University Timetabling Problem. In: Proceedings on the 1st ACM/SIGEVO Summit on Genetic and Evolutionary Computation, 2009.

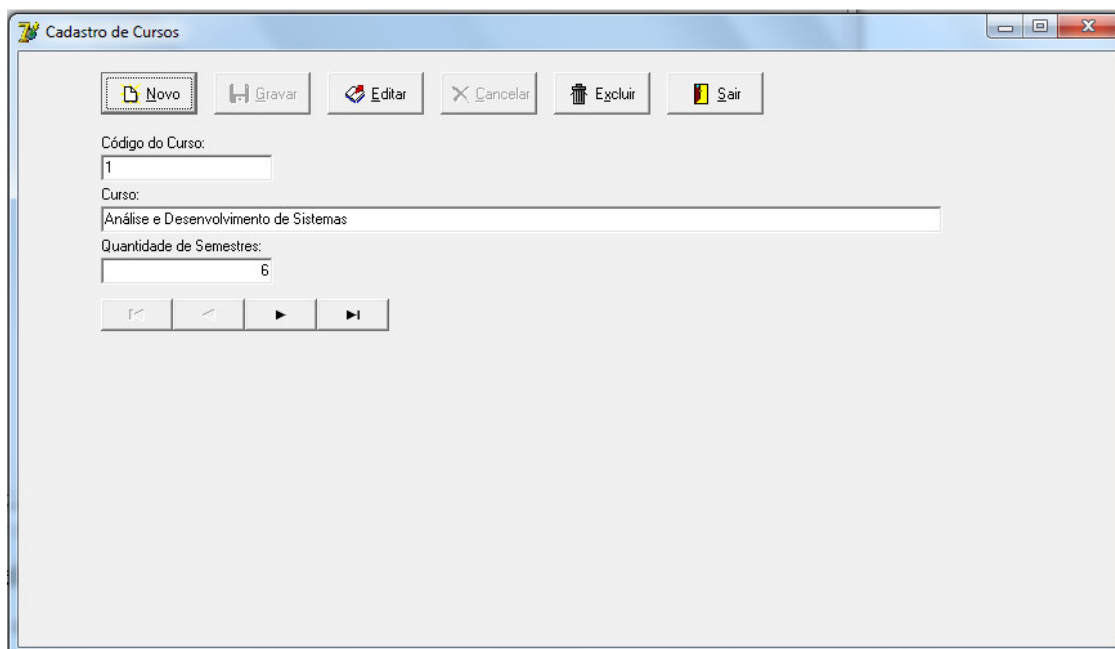
ANEXO A: MANUAL DO SISTEMA

Neste anexo, são apresentadas as principais funcionalidades de cada uma das telas do sistema. O sistema possui, além das telas de geração da grade horária e ajuste de parâmetros, opções de cadastro e de geração de relatórios que serão apresentadas nesse tópico.

O módulo de cadastro está dividido em quatro partes: o cadastro de cursos, de períodos, de professores e de disciplinas. Esse módulo será acessado pelo coordenador de curso e representa as primeiras ações a serem executadas para a geração da grade horária.

A.1 – Cadastro de Cursos

O cadastro de cursos pode ser acessado na tela principal do sistema, clicando na opção Cadastros e depois na opção Cursos. Nele, são realizadas as operações referentes aos cursos oferecidos pela instituição. Nesse cadastro é possível inserir novos cursos, alterar informações dos cursos já cadastrados e excluir cursos. As informações armazenadas sobre um curso são o código do curso, o nome do curso e a quantidade de semestres que ele possui. O cadastro de cursos é visualizado na figura A.1 mostrada a seguir. Para esse e para todos os outros cadastros existentes no sistema, para realizar um novo cadastro, é necessário clicar no botão Novo, preencher os campos existentes e clicar no botão Gravar. Para realizar a alteração em um item já cadastrado, é necessário clicar no botão Editar, realizar as alterações e clicar no botão Gravar quando todas elas tiverem sido realizadas. Também é possível cancelar as alterações antes que elas tenham sido salvas, para isso, basta clicar no botão Cancelar. Para excluir um item cadastrado, basta localizá-lo e clicar no botão Excluir. É possível navegar pelas informações já cadastradas através da barra de navegação existente no canto inferior da tela, que possui botões que possibilitam ir para o primeiro registro, para o anterior, para o próximo ou para o último, respectivamente.



The image shows a software window titled "Cadastro de Cursos". At the top, there is a toolbar with six buttons: "Novo" (with a document icon), "Gravar" (with a floppy disk icon), "Editar" (with a pencil icon), "Cancelar" (with an 'X' icon), "Excluir" (with a trash can icon), and "Sair" (with a door icon). Below the toolbar, there are three input fields. The first is labeled "Código do Curso:" and contains the number "1". The second is labeled "Curso:" and contains the text "Análise e Desenvolvimento de Sistemas". The third is labeled "Quantidade de Semestres:" and contains the number "6". At the bottom of the window, there are four navigation buttons: a left arrow, a right arrow, a play button, and a stop button.

Figura A.1 – Cadastro de cursos

A.2 – Cadastro de Períodos

O cadastro de períodos pode ser acessado na tela principal do sistema, clicando na opção Cadastros e depois na opção Períodos. Nele, são realizadas as operações referentes aos períodos de cada um dos cursos oferecidos pela instituição. Nesse cadastro, para cada curso é possível inserir novos períodos, alterar informações dos períodos já cadastrados e excluir períodos. As informações armazenadas sobre um período são o nome de um curso já cadastrado, o nome do período e a quantidade de aulas que ele possui. Em relação à quantidade de aulas, deve-se obedecer ao limite máximo de 25 aulas semanais, pois os cursos da instituição que se baseia esse estudo de caso são oferecidos no período noturno com 5 aulas diárias de segunda a sexta-feira. O cadastro de períodos é visualizado na figura A.2 mostrada a seguir.

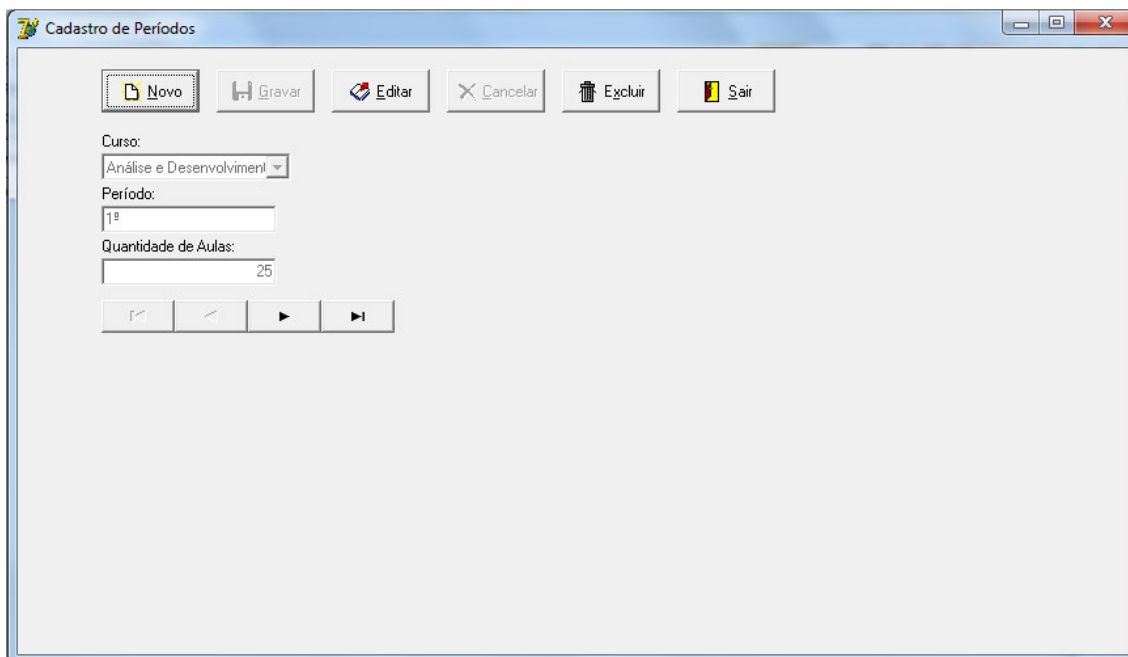


Figura A.2 – Cadastro de períodos

A.3 - Cadastro de Professores

O cadastro de professores pode ser acessado na tela principal do sistema, clicando na opção Cadastros e depois na opção Professores. Nele, são realizadas as operações referentes aos professores da instituição. Nesse cadastro, é possível inserir novos professores, alterar informações sobre os professores já cadastrados e excluir professores. As informações armazenadas sobre cada professor são o código no SIAPE, o nome completo do professor, se ele possui dedicação exclusiva na instituição e a sua disponibilidade para ministrar aulas em cada um dos horários em cada um dos dias da semana.

A disponibilidade de um professor pode ser cadastrada como disponível, indisponível ou indesejado para cada um dos horários da semana. Horário disponível é o horário em que o professor pode ministrar aulas, horário indisponível é o horário em que o professor não pode ministrar aulas, enquanto horário indesejado seria um horário no qual o professor pode ministrar aulas, porém prefere não ministrar.

Para alterar a disponibilidade de um professor no sistema, é necessário localizá-lo, caso seja um professor já cadastrado, clicar no botão Editar e alterar individualmente o *status* de cada um dos horários de aula semanais. Para efetuar a alteração, basta clicar no símbolo referente ao horário que se deseja alterar no quadro de disponibilidade, de acordo com a legenda apresentada do seu lado direito. Depois de efetuadas todas as alterações, basta clicar no botão Gravar. Quando se cadastra um novo professor, inicialmente, todos os seus horários são inicializados com o *status* de disponível. Clicando uma vez, o *status* é alterado para indisponível. Clicando outra vez, ele é alterado para indesejado.

Clicando outra vez, ele retorna para disponível. O cadastro de professores é visualizado na figura A.3 mostrada a seguir.

	Segunda	Terça	Quarta	Quinta	Sexta
19:00	✓	✓	✓	⚠	✗
19:45	✓	✓	✓	⚠	✗
20:30	✓	✓	✓	⚠	✗
21:30	✓	✓	✗	⚠	✗
22:15	✓	✓	✗	⚠	✗

Figura A.3 – Cadastro de professores

A.4 – Cadastro de Disciplinas

O cadastro de disciplinas pode ser acessado na tela principal do sistema, clicando na opção Cadastros e depois na opção Disciplinas. No cadastro de disciplinas são realizadas as operações referentes às disciplinas oferecidas em cada um dos períodos dos cursos cadastrados. Nesse cadastro, é possível inserir novas disciplinas, alterar informações das disciplinas já cadastradas e excluir disciplinas. As informações armazenadas sobre uma disciplina são o curso já cadastrado a que ela pertence, o período já cadastrado a que ela pertence, o código da disciplina, o nome da disciplina, o professor já cadastrado que será o responsável pela disciplina, a quantidade de aulas semanais que a disciplina possui e a divisão dessas aulas.

A divisão das aulas é feita de acordo com a preferência do professor responsável pela disciplina. Em um primeiro momento, deve ser realizada uma divisão entre a quantidade de aulas teóricas e práticas da disciplina. Em um segundo momento, deve ser realizada uma divisão entre a quantidade de aulas simples, duplas ou triplas, de aulas teóricas ou práticas.

O sistema faz a validação para que, obrigatoriamente, a quantidade de aulas semanais seja o somatório entre as aulas teóricas e práticas. Ele também realiza a validação para que, a quantidade de aulas teóricas seja o somatório das aulas individuais, duplas e triplas teóricas. A mesma validação existe para a quantidade de aulas práticas, que equivale ao somatório das aulas individuais, duplas e triplas práticas. Caso alguma dessas validações não seja cumprida, o sistema informa através de uma mensagem, a necessidade de correção. Só é possível gravar a disciplina, caso essas validações sejam respeitadas.

Também podem ser cadastradas aulas de uma disciplina que são obrigatórias de serem ministradas em algum horário específico de algum dos dias da semana. Essas aulas podem ser obrigatórias por causa da disponibilidade de um determinado laboratório, recurso, professor, dentre outros. No sistema, um horário obrigatório pode ser cadastrado como teórico ou prático, para cada um dos horários da semana. Para colocar uma determinada aula de uma determinada disciplina como obrigatória de ser realizada em determinado horário, é necessário localizar a disciplina, clicar no botão Editar e alterar individualmente o *status* de cada um dos horários de aula semanais que se pretende colocar como obrigatório. Para isso, basta clicar no símbolo referente ao horário que se deseja alterar o *status* na tabela de horários obrigatórios, de acordo com a legenda apresentada do seu lado direito. Depois de realizada a alteração, basta clicar no botão Gravar. Quando se cadastra uma nova disciplina, inicialmente, todos os seus horários ficam com o *status* de Qualquer Aula, ou seja, sem aulas obrigatórias, podendo ser as aulas da disciplina distribuídas em qualquer horário da semana. Caso se deseje alterar, criando um horário obrigatório, é necessário clicar uma vez no horário que se deseja alterar, o *status* desse horário é então alterado para Aula Prática, indicando que nesse horário obrigatoriamente deverá ser realizada uma aula prática dessa disciplina. Clicando outra vez, ele é alterado para Aula Teórica, indicando que nesse horário obrigatoriamente deverá ser realizada uma aula teórica dessa disciplina. Clicando outra vez, o *status* volta para Qualquer Aula.

É importante destacar que o sistema não permite que sejam cadastradas aulas obrigatórias de mais de uma disciplina de um mesmo período em um mesmo horário, fato que impossibilitaria a criação de uma grade horária válida. Caso ocorra essa situação, o sistema mostra uma mensagem informando o erro, para que seja realizada a correção. Só é possível gravar a disciplina, caso essa validação seja respeitada.

O cadastro de professores é visualizado na figura A.4 mostrada a seguir.

Cadastro de Disciplinas

Curso:

Período:

Código da disciplina:

Disciplina:

Professor:

Quantidade de Aulas Semanas:

Quantidade de Aulas Teóricas:
 Aulas Triplas Teóricas:
 Aulas Duplas Teóricas:
 Aulas Simples Teóricas:

Quantidade de Aulas Práticas:
 Aulas Triplas Práticas:
 Aulas Duplas Práticas:
 Aulas Simples Práticas:

	Segunda	Terça	Quarta	Quinta	Sexta
19:00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19:45	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20:30	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21:30	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
22:15	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Qualquer Aula
 Aula Teórica
 Aula Prática

Figura A.4 – Cadastro de professores

A.5 – Relatório completo

O relatório completo é um relatório que apresenta toda a grade horária do curso. Ele mostra para cada período, quais são as disciplinas que serão ministradas e quem é o professor responsável por cada uma delas, em cada dia e horário da semana.

A figura A.5 a seguir mostra um exemplo de relatório completo gerado pelo sistema. A figura mostra apenas os dois primeiros períodos, embora o sistema gere o relatório de todos os períodos do curso.

1º Período

	Segunda-feira	Terça-feira	Quarta-feira	Quinta-feira	Sexta-feira
19:00	ALGEB - T José Ricardo	ALGOR - T Ermani Melo	ALGOR - P Ermani Melo	SI - P Hugo Leonardo	ALGEB - T José Ricardo
19:45	ALGOR - P Ermani Melo	ALGOR - T Ermani Melo	ALGOR - P Ermani Melo	PORT - T Tamara	ATIV - P Paula Nakamoto
20:30	ALGOR - P Ermani Melo	PDI - P Alexandre Ribeiro	ING - T Sandra Gardelari	SI - P Hugo Leonardo	SI - T Hugo Leonardo
21:30	PRO - T Marcelo Ponciano	PRO - T Marcelo Ponciano	ING - T Sandra Gardelari	ALGEB - T José Ricardo	ALGOR - T Ermani Melo
22:15	PDI - P Alexandre Ribeiro	PRO - T Marcelo Ponciano	PORT - T Tamara	ATIV - P Paula Nakamoto	ALGOR - T Ermani Melo

2º Período

	Segunda-feira	Terça-feira	Quarta-feira	Quinta-feira	Sexta-feira
19:00	ESTR - P Rafael Godói	MOD - T Hugo Leonardo	EMPR - T Vicente	PROB - T José Ricardo	ESTR - T Rafael Godói
19:45	ESTR - T Rafael Godói	MOD - P Hugo Leonardo	FM - T José Ricardo	ESTR - P Rafael Godói	ARQ - T Marcelo Barreiro
20:30	ESTR - P Rafael Godói	FM - T José Ricardo	ESOF1 - T Willian Manzan	ESOF1 - T Willian Manzan	ARQ - T Marcelo Barreiro
21:30	FM - T José Ricardo	EMPR - T Vicente	ARQ - T Marcelo Barreiro	ESTR - T Rafael Godói	ESOF1 - T Willian Manzan
22:15	FM - T José Ricardo	PROB - T José Ricardo	ARQ - T Marcelo Barreiro	MOD - T Hugo Leonardo	MOD - P Hugo Leonardo

Figura A.5 – Relatório completo da grade horária

A.6 – Relatório por professores

O relatório por professores mostra uma relação de todos os professores cadastrados no sistema. Para cada professor é mostrada uma grade horária com as disciplinas que ele irá ministrar e para qual período as aulas serão ministradas, em cada dia e horário da semana.

A figura A.6 a seguir mostra um exemplo de relatório por professores gerado pelo sistema. A figura mostra apenas dois professores, embora o sistema gere o relatório de todos os professores do curso.

Hugo Leonardo

	Segunda-feira	Terça-feira	Quarta-feira	Quinta-feira	Sexta-feira
19:00	----- -----	PESQ - T 3º Período	----- -----	SI - P 1º Período	SI - T 1º Período
19:45	----- -----	PESQ - T 3º Período	----- -----	----- -----	SI - P 1º Período
20:30	----- -----	PESQ - T 3º Período	----- -----	MOD - T 2º Período	----- -----
21:30	----- -----	MOD - P 2º Período	----- -----	PESQ - P 3º Período	MOD - T 2º Período
22:15	----- -----	----- -----	----- -----	PESQ - P 3º Período	MOD - P 2º Período

Jéfferson Beethoven

	Segunda-feira	Terça-feira	Quarta-feira	Quinta-feira	Sexta-feira
19:00	----- -----	PDM - T 5º Período	----- -----	PDM - T 6º Período	----- -----
19:45	----- -----	PDM - T 5º Período	----- -----	PDM - T 6º Período	PDM - P 5º Período
20:30	----- -----	----- -----	----- -----	COM - T 6º Período	----- -----
21:30	PDM - P 5º Período	COM - T 6º Período	COM - T 6º Período	----- -----	PDM - P 6º Período
22:15	PDM - P 5º Período	PDM - T 6º Período	COM - T 6º Período	----- -----	PDM - P 6º Período

Figura A.6 – Relatório da grade horária por professores

ANEXO B: QUESTIONÁRIO PARA COLETA DE REQUISITOS

INFORMAÇÕES PARA ELABORAÇÃO DA GRADE HORÁRIA PARA O 2º SEMESTRE DO ANO DE 2012 – CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Nome do professor: _____ SIAPE: _____

Disponibilidade:

	Segunda-feira	Terça-feira	Quarta-feira	Quinta-feira	Sexta-feira
19:00					
19:45					
20:30					
21:30					
22:15					

Favor assinalar na tabela acima para cada um dos horários, os seguintes códigos:

- 0, se o professor estiver disponível no horário
- 1, se o professor estiver indisponível no horário
- 2, se o professor estiver disponível no horário, porém não desejar ministrar aula nele

Disciplinas ministradas:

Nome da disciplina: _____ Período: _____

Quantidade de aulas semanais: _____

Essas aulas são divididas em: _____ Aulas teóricas _____ Aulas práticas

O professor prefere ministrar as aulas teóricas em:

_____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

O professor prefere ministrar as aulas práticas em:

_____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

Disciplinas ministradas:

Nome da disciplina: _____ Período: _____

Quantidade de aulas semanais: _____

Essas aulas são divididas em: _____ Aulas teóricas _____ Aulas práticas

O professor prefere ministrar as aulas teóricas em:

_____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

O professor prefere ministrar as aulas práticas em:

_____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

Disciplinas ministradas:

Nome da disciplina: _____ Período: _____

Quantidade de aulas semanais: _____

Essas aulas são divididas em: _____ Aulas teóricas _____ Aulas práticas

O professor prefere ministrar as aulas teóricas em:

_____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

O professor prefere ministrar as aulas práticas em:

_____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

Disciplinas ministradas:
 Nome da disciplina: _____ Período: _____
 Quantidade de aulas semanais: _____
 Essas aulas são divididas em: _____ Aulas teóricas _____ Aulas práticas
 O professor prefere ministrar as aulas teóricas em:
 _____ Aulas individuais _____ Aulas duplas _____ Aulas triplas
 O professor prefere ministrar as aulas práticas em:
 _____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

Disciplinas ministradas:
 Nome da disciplina: _____ Período: _____
 Quantidade de aulas semanais: _____
 Essas aulas são divididas em: _____ Aulas teóricas _____ Aulas práticas
 O professor prefere ministrar as aulas teóricas em:
 _____ Aulas individuais _____ Aulas duplas _____ Aulas triplas
 O professor prefere ministrar as aulas práticas em:
 _____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

Disciplinas ministradas:
 Nome da disciplina: _____ Período: _____
 Quantidade de aulas semanais: _____
 Essas aulas são divididas em: _____ Aulas teóricas _____ Aulas práticas
 O professor prefere ministrar as aulas teóricas em:
 _____ Aulas individuais _____ Aulas duplas _____ Aulas triplas
 O professor prefere ministrar as aulas práticas em:
 _____ Aulas individuais _____ Aulas duplas _____ Aulas triplas

Alguma das disciplinas ministradas precisa obrigatoriamente ser ministrada em um horário específico?
 Se sim, qual é a disciplina e qual é esse horário?

Uberaba, ____ de _____ de 2012.

 Assinatura do professor