

UNIVERSIDADE SAGRADO CORAÇÃO

CLAUDENIR DE FREITAS MACHADO

**GERAÇÃO AUTOMÁTICA DE
HORÁRIOS ESCOLARES
UTILIZANDO ALGORITMOS DE
SATISFAÇÃO DE RESTRIÇÕES**

**BAURU
2013**

CLAUDENIR DE FREITAS MACHADO

**GERAÇÃO AUTOMÁTICA DE
HORÁRIOS ESCOLARES
UTILIZANDO ALGORITMOS DE
SATISFAÇÃO DE RESTRIÇÕES**

Trabalho de Conclusão de Curso,
apresentado ao Centro de Exatas e Sociais
Aplicadas como parte dos requisitos para
obtenção do título em Bacharel em Ciência da
Computação, sob orientação do Prof. Ms.
Patrick Pedreira Silva.

**BAURU
2013**

Machado, Claudenir de Freitas
M1491g

Geração automática de horários escolares utilizando algoritmos de satisfação de restrições / Claudenir de Freitas Machado -- 2013.
65f. : il.

Orientador: Prof. Me. Patrick Pedreira Silva.
Coorientador: Prof. Me. Márcio Cardim.

Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade Sagrado Coração – Bauru – SP.

1. Inteligência artificial, IA. 2. Problemas de satisfação de restrições, PSR. 3. Problema de quadros escolares. I. Silva, Patrick Pedreira. II. Cardim, Márcio. III. Título.

CLAUDENIR DE FREITAS MACHADO

**GERAÇÃO AUTOMÁTICA DE HORÁRIOS ESCOLARES
UTILIZANDO ALGORITMOS DE SATISFAÇÃO DE RESTRIÇÕES**

Trabalho de Conclusão de Curso, apresentado ao Centro de Exatas e Sociais aplicadas da Universidade Sagrado Coração como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação sob orientação do Prof. Ms. Patrick Pedreira Silva.

Banca Examinadora:

Prof. Ms. Patrick Pedreira Silva

Universidade Sagrado Coração

Prof. Ms. Márcio Cardim

Universidade Sagrado Coração

Prof. Dr. Elvio Gilberto Silva

Universidade Sagrado Coração

Bauru, 06 de Dezembro de 2013.

Dedico este trabalho a todos que me apoiaram no decorrer do curso, em especial a Dona Sandra e a senhorita Andressa.

AGRADECIMENTOS

Primeiramente a Deus, minha família, meus amigos e colegas de cursos que sempre me incentivaram, aos professores, ao professor e orientador Patrick por colaborar no andamento deste, senhorita Andressa em especial pelo incentivo e a todos aqueles que de uma forma direta ou indireta influenciaram no decorrer do mesmo.

RESUMO

Atualmente diversas instituições de ensino enfrentam uma grande dificuldade para geração de horários escolares, onde em nível de complexidade envolve diversas variáveis dentro de um contexto realizado de forma manual. Desta forma, um problema com tais características exige, para a sua resolução, métodos para decomposição e busca de soluções. Analisando tal situação, a automatização do processo de geração de horário escolar pode trazer benefícios, justificando, deste modo, investigações nesta área. Partindo deste princípio e utilizando o conceito de Problemas de Satisfação de Restrições (PSR), esta pesquisa visou o desenvolvimento de uma ferramenta que tratou todas as informações pertinentes ao problema citado, a fim de gerar automaticamente os horários, respeitando a cada restrição imposta ao longo do processo. A ferramenta foi desenvolvida considerando um ambiente específico de uma Universidade e curso particular, demonstrando que o Algoritmo de Satisfação de Restrições (PSR) é eficaz no ato da geração de horários escolares.

Palavras-chave: Inteligência Artificial. IA. Problema de Satisfação de Restrições. PSR. Problema geração de horários.

ABSTRACT

Currently several educational institutions face a major difficulty for the generation of school timetables, where level of complexity involves several variables among a context performed manually. Thus, a problem with such features requires for its resolution, methods for decomposition and search for solutions. Analyzing this situation, automating the generation of school time process can bring benefits, justifying thus research in this area. With this assumption and using the concept of Constraint Satisfaction Problems (PSR), this research aimed at developing a tool that addressed all relevant to the problem mentioned in order to automatically generate schedules respecting each restriction imposed over information process. The tool was developed considering a particular environment for a particular course and university, demonstrating that the algorithm Constraint Satisfaction (PSR) is effective upon the generation of school timetables.

Keywords: Artificial Intelligence. IA. Constraint Satisfaction Problem. PSR. Problem generating schedules.

LISTA DE FIGURAS

Figura 1- Algumas definições de Inteligência Artificial, organizadas em quatro categorias.	9
Figura 2 - Cérebro virtual que simula a força do pensamento.	13
Figura 3 - Busca em profundidade em uma árvore binária.....	22
Figura 4 - Busca em extensão em uma árvore binária simples	22
Figura 5 - Estados e territórios da Austrália.	24
Figura 6 - Solução para o problema de coloração dos mapas.....	25
Figura 7 - Parte da árvore de busca utilizando o retrocesso simples.	26
Figura 8 - A coloração azul discrimina o estado com maior grau.	26
Figura 9 - Fluxo da Informação.	29
Figura 10 - Hierarquia para Geração do Horário.	31
Figura 11 - Lista com a relação de Professores.....	31
Figura 12 - Objetos carregados com as informações dos Professores.	32
Figura 13 - Algoritmo [1].	33
Figura 14 - Algoritmo [2]	34
Figura 15 - Algoritmo [3].	34
Figura 16 - Algoritmo [4]	35
Figura 17 - Algoritmo [5].	35
Figura 18 - Horário Ciência da Computação 2º Semestre.....	36
Figura 19 - Modelo Entidade Relacionamento (MER).....	36
Figura 20 - Estrutura geral da aplicação.....	39
Figura 21 - Tela Inicial Aplicação.....	41
Figura 22 - Cadastro de Cursos.....	42
Figura 23 - Cadastro de Disciplinas.	42
Figura 24 - Cadastro de Cursos – Semestre.....	43
Figura 25 - Cadastro de Instituições.....	43
Figura 26 - Cadastro de Professores.	44
Figura 27 - Cadastro de Professores, vínculo com as disciplinas.....	45
Figura 28 - Cadastro de Professores - Disponibilidade.	45
Figura 29 - Geração de Horário.....	46
Figura 30 - Horários gerados.	46
Figura 31 - Configurações de Notificação.....	47
Figura 32 - Configurações da aplicação.....	48
Figura 33 - Sobre. Fonte: Autor.	48
Figura 34 - Relação Dados Professor.....	49
Figura 35 - Relação Dados Curso - Disciplina.....	51
Figura 36 - Relação Professor-Disciplina.	54
Figura 37 - Geração de Horário. 1ª Etapa.....	55
Figura 38 - Geração de Horário. 2ª Etapa.....	55
Figura 39 - Geração de Horário. 3ª Etapa.....	56
Figura 40 - Geração de Horário. 4ª Etapa.....	57
Figura 41 - Geração de Horário. 5ª Etapa.....	57
Figura 42 - Geração de Horário. 6ª Etapa.....	58

Figura 43 - Geração de Horário. 7ª Etapa.....	58
Figura 44 - Geração de Horário. 8ª Etapa.....	59
Figura 45 - Horário Final. 9ª Etapa.....	59
Figura 46 - Análise disciplinas 'fixas'.....	60
Figura 47 - Geração Horário. 9ª Etapa.....	61
Figura 48 - Geração Horário. 10ª Etapa.....	61
Figura 49 - Geração Horário. 11ª Etapa.....	62
Figura 50 - Geração Horário. 12ª Etapa.....	63
Figura 51 - Horário Final. 13ª Etapa.....	63

SUMÁRIO

AGRADECIMENTOS	6
1 INTRODUÇÃO	6
2 OBJETIVOS	8
2.1 OBJETIVO GERAL.....	8
2.2 OBJETIVOS ESPECÍFICOS	8
3 INTELIGÊNCIA ARTIFICIAL	9
3.1 INTRODUÇÃO	9
3.2 MODELOS DE INTELIGÊNCIA ARTIFICIAL.....	17
3.3 INTELIGÊNCIA E CONHECIMENTO	18
3.3.1 REPRESENTAÇÃO DO CONHECIMENTO	20
3.4 MÉTODOS DE BUSCA E REPRESENTAÇÃO DO CONHECIMENTO	21
3.5 RESTRIÇÕES NA RESOLUÇÃO DE PROBLEMAS	23
3.5.1 PROBLEMA DE SATISFAÇÃO DE RESTRIÇÕES.....	24
4 PROBLEMA DE GERAÇÃO DE HORÁRIO ESCOLAR NA UNIVERSIDADE SAGRADO CORAÇÃO.....	27
5 METODOLOGIA	29
5.1 DEFINIÇÃO DO ALGORITMO DE GERAÇÃO DE HORÁRIOS.....	29
5.2 DESENVOLVIMENTO	36
6 AVALIAÇÃO DO SISTEMA.....	49
7 CONSIDERAÇÕES FINAIS.....	64
REFERÊNCIAS	65

1 INTRODUÇÃO

Segundo Rich e Knight (1994), "IA é a área da Ciência da Computação orientada ao entendimento, construção e validação de sistemas inteligentes, isto é, que exibem de alguma forma, características associadas ao que chamamos inteligência".

A Inteligência Artificial (IA) é uma disciplina relacionada à Computação que desperta muito interesse por ser tratar de uma ciência recente e que abrange diversas áreas.

"Não é meu objetivo surpreendê-los ou chocá-los - mas o modo mais simples de resumir tudo isso é dizer que agora existem no mundo máquinas que pensam, aprendem e criam. Além disso, sua capacidade de realizar essas atividades está crescendo rapidamente até o ponto - em um futuro visível - no qual a variedade de problemas com que elas poderão lidar será correspondente à variedade de problemas com os quais lida a mente humana." Russell (2004).

Nos dias atuais, IA faz parte de nossas vidas desde celulares com funcionalidades diversas até mesmo robôs trazem benefícios à medicina e áreas afins. Dentre diversos tópicos relacionados à área, aqueles que lidam com os chamados problemas de satisfação de restrições permitem utilizar técnicas que visam simplificar a forma como esta classe de problemas pode ser resolvida. Os algoritmos aplicados na resolução destes problemas permitem modelá-los por meio de estados que obedecem a uma representação padrão, estruturada e muito simples, tornando mais fácil resolver problemas extensos. Tais algoritmos auxiliam no desenvolvimento de ferramentas que manipulam uma série de restrições, com o objetivo de chegar a um determinado resultado, respeitando cada condição imposta.

Segundo definição de Russell (2004), um Problema de Satisfação de Restrições (PSR) é definido por um conjunto de variáveis e restrições. Cada variável tem um domínio não vazio de possíveis valores e cada restrição envolve algum subconjunto das variáveis, e especifica as combinações de valores permitidas para aquele subconjunto. Vários problemas podem ser

considerados como um problema de satisfação de restrições como, por exemplo, o problema de montagem de um horário escolar.

Atualmente, diversas Instituições de ensino possuem uma grande dificuldade com relação à geração de horários escolares. Visto que, para gerar estes horários manualmente é necessário respeitar uma série de restrições (disponibilidade de cada professor, a disciplina que condiz com o semestre vigente, a quantidade de turmas e a relação entre professor-disciplina). Desta forma, um problema com tais características exige, para a sua resolução, métodos para decomposição e busca de soluções.

Neste contexto, a automatização do processo de geração de horário escolar pode trazer benefícios, justificando, deste modo, investigações nesta área. Partindo deste princípio e utilizando o conceito de PSR, esta pesquisa visou o desenvolvimento de uma ferramenta que tratou todas as informações pertinentes ao problema citado, a fim de gerar os horários, evitando lacunas, respeitando a cada restrição imposta ao longo do processo. A ferramenta foi desenvolvida considerando um ambiente específico de uma universidade e curso particular. Todo o processo de concepção da ferramenta e seus algoritmos seguiram como eixo norteador para o processo de geração de horário para o curso de Ciência da Computação.

Assim, a proposta foi criar uma ferramenta com uma *interface* interativa entre homem-máquina, onde a informação seja de fácil manipulação e que atenda a todas as necessidades impostas.

Este trabalho está estruturado da seguinte forma: o capítulo 2 traz os objetivos (gerais e específicos); o capítulo 3 aborda os conceitos relacionados à área de Inteligência Artificial; o capítulo 4 aborda a grande dificuldade de geração de horários escolares; o capítulo 5 aborda a metodologia; o capítulo 6 aborda os resultados e por fim, as considerações finais.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Desenvolver uma ferramenta de geração automática de horários escolares, utilizando o conceito de algoritmos para Problemas de Satisfação de Restrições.

2.2 OBJETIVOS ESPECÍFICOS

- Abordar o conceito de Inteligência Artificial inicialmente, focando sua real importância neste projeto;
- Desenvolver um algoritmo com PSR que satisfaça as restrições impostas, tais como a disponibilidade do professor, as disciplinas, a quantidade de turmas e a relação entre professor-disciplina;
- Desenvolver uma aplicação com uma *interface* interativa, onde o usuário tenha acesso fácil para manipular as informações;
- Realizar levantamento de dados com os resultados obtidos;
- Testar o sistema a fim de verificar sua efetividade para a tarefa proposta.

3 INTELIGÊNCIA ARTIFICIAL

3.1 INTRODUÇÃO

A Inteligência Artificial (IA) é considerada uma ciência recente, já que seu surgimento se deu logo após a Segunda Guerra Mundial, em meados de 1956. Dentre várias definições, podemos definir que a IA é voltada para área de desenvolvimento de ferramentas inteligentes, ou seja, que possuem comportamentos voltados aos aspectos humanos. Na tabela a seguir, seguem algumas definições de IA:

Sistemas que pensam como seres humanos
“O novo e interessante esforço para fazer os computadores pensarem... máquinas em com mentes, no sentido total e literal.” (Haugeland, 1985)
“[Automatização de] atividades que associamos ao pensamento humano, atividades como a tomada de decisões, a resolução de problemas, o aprendizado...” (Bellman, 1978)
Sistemas que atuam como seres humanos
“A Arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas.”(Kurzweil, 1990)
Sistemas que pensam racionalmente
“O estudo das faculdades mentais pelo uso de modelos computacionais.” (Charniak e McDermott, 1985)
“O estudo das computações que tornam possível perceber, raciocinar e agir.”(Winston, 1992)
Sistemas que atuam racionalmente
“A Inteligência Artificial é o estudo do projeto de agentes inteligentes.” (Pool et al., 1998)

Figura 1- Algumas definições de Inteligência Artificial, organizadas em quatro categorias. Fonte: Russell (2004).

Na Figura 1, os dois primeiros tópicos abordam citações voltadas ao desempenho humano enquanto os demais abordam citações voltadas à inteligência e racionalidade. Segundo Boose (1994), a Inteligência Artificial

busca entender o funcionamento da mente humana e, através de outros mecanismos, imitá-la. Para entender o funcionamento da mente, uma série de perguntas está relacionada ao tema, tais como: Como ocorre o pensamento? Como o homem extrai conhecimento do mundo? Como a memória, os sentidos e a linguagem ajudam no desenvolvimento da inteligência? Como surgem as ideias? Como a mente processa informações e tira conclusões decidindo por uma coisa ao invés de outra?

Os estudos de IA têm seguido quatro estratégias, sendo elas divididas em abordagens centradas em torno de seres humanos e abordagens centradas em torno da racionalidade. Segundo Russell (2004), uma abordagem centrada nos seres humanos deve ser uma ciência empírica, envolvendo hipóteses e confirmação experimental e, uma abordagem racionalista envolve uma combinação de matemática e engenharia.

Segundo Simon (1988), ele aponta a IA em duas abordagens:

- **Abordagem Cognitiva:** forma de como o ser humano raciocina, com o objetivo de encontrar uma melhor explicação para comportamentos inteligentes baseados em aspectos psicológicos e processos algorítmicos;
- **Abordagem Conexionista:** dá ênfase no modelo de funcionamento do cérebro, dos neurônios e das conexões neurais.

Atualmente, a IA abrange diversas áreas tornando-a um campo universal de estudos e pesquisas. Segundo Russell (2004), a “IA sistematiza e automatiza tarefas intelectuais e, portanto, é potencialmente relevante para qualquer esfera da atividade intelectual humana”.

Os fundamentos da Inteligência Artificial só foram possíveis mediante a contribuição de diversas áreas, onde estas áreas contribuíram com ideias, pontos de vista e técnicas. Segue um breve conceito de cada área:

- **Filosofia**

Grandes filósofos demonstraram através de conceitos como que a mente humana se desenvolve a partir do cérebro físico e como que o

conhecimento conduz a ação. Esta última questão da conexão entre conhecimento e ação é de grande importância para a IA visto que a inteligência exige ação, com base em um raciocínio. Para Aristóteles, as ações se baseavam num conceito lógico entre metas e o conhecimento do resultado da ação. Um detalhe interessante deste tópico foi uma obra de Aristóteles, *Ética a Nicômico*, onde o mesmo sugere um algoritmo e depois de passados 2300 anos, Newell e Simon em um programa GPS.

- **Matemática:**

A lógica, a computação e a probabilidade foram às três áreas fundamentais para formalização matemática.

A ideia de lógica formal teve início com o trabalho de George Boole (1815-1864), onde o mesmo definiu os detalhes da lógica booleana ou lógica proposicional. Logo depois Gottlob Frege (1848-1925) realizou algumas alterações para inclusão de objetos e relações, criando a lógica de primeira ordem, que atualmente é utilizado como meio básico para representação do conhecimento. Não muito distante Alfred Tarski (1902-1983) estudou uma forma de relacionar os objetos de uma lógica a objetos do mundo real.

Diversos matemáticos contribuíram na relação entre lógica e computação, onde aparentemente o algoritmo de Euclides para cálculo do maior denominador comum tenha dado o pontapé inicial. Logo depois, Boole e outros formalizaram o raciocínio matemático geral como dedução lógica. Não muito distante David Hilbert (1862-1943) listou uma série de problemas, onde a última perguntava se era possível existir um algoritmo para definir a verdade de qualquer proposição lógica relacionado a números naturais. Mediante a esta situação surgiu o teorema da incompleteza de Kurt Gödel (1906-1978), onde definiu que algumas funções não podem ser calculadas, porém, Allan Turing (1912-1954) se sentiu motivado a provar o oposto. Segundo Fernandes (2003), a tese de Church-Turing, afirma que a máquina de Turing é capaz de calcular qualquer função computável.

A questão da probabilidade teve início quando Gerolamo Cardano (1501-1576) descreveu em termos os resultados possíveis de um jogo de azar, partindo deste ponto, diversos outros pesquisadores melhoraram a teoria e desenvolveram novos métodos estatísticos.

- **Economia:**

Adam Smith (1723-1790) dentre muitos filósofos, foi o primeiro a tratar a economia como uma ciência, considerando que as economias podem maximizar o bem estar econômico. A teoria da decisão fornece uma estrutura formal e completa para tomadas de decisões em situações de incertezas, onde é mais apropriado para grandes economias, em que a ação de um agente não implica na ação de outro. Entretanto, no caso de pequenas economias as ações de um agente implicam na ação de outros agentes, parecido com um jogo. A teoria dos jogos de Von Neumann e Morgenstern mostrou um agente racional deve agir de forma casual ou pelo menos de uma forma que pareça casual para os adversários. A economia contribuiu muito para noção de agentes racionais.

- **Neurociência:**

A neurociência é o estudo científico do sistema nervoso, voltado mais para a área do cérebro, local onde se observou que o pensamento está associado ao mesmo. Ainda que existam diversos estudos relacionados ao funcionamento do cérebro, não foi definida nenhuma teoria que compreenda como a memória de cada indivíduo é armazenada. Segundo definição de (Searle, 1992), “o cérebro é uma coleção de células simples que pode levar ao pensamento, à ação e à consciência ou, em outras palavras, que cérebros geram mentes”. Outra linha é a teoria do misticismo, que diz que as mentes operam em um reino que esta além da ciência física.

A neurociência computacional é a área da neurociência que tem como finalidade propor modelos matemáticos e computacionais para representar o sistema nervoso. Moraes (2012) abordou o desenvolvimento de Chris Eliasmith (2012) que consiste em um cérebro virtual que simula comportamentos e que realiza tarefas cognitivas, o modelo denominado Spaun é formado por 2,5 milhões de neurônios virtuais organizados em sistemas conectados, comparados a cada partição do cérebro humano. Segue uma representação na Figura 2:

FORÇA DO PENSAMENTO

Cérebro totalmente virtual pode realizar tarefas básicas de um teste de QI

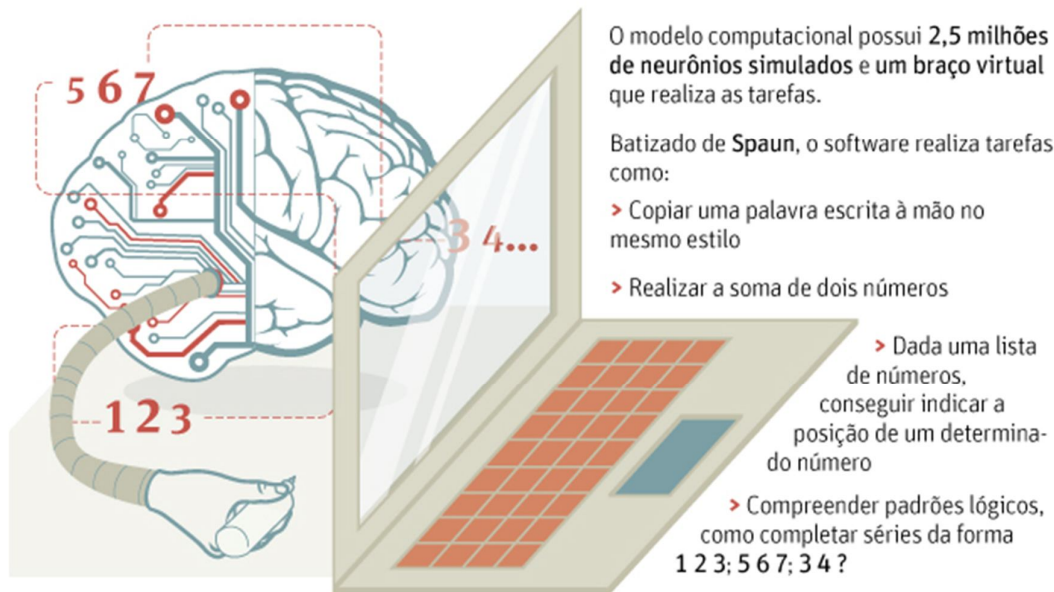


Figura 2 - Cérebro virtual que simula a força do pensamento.

Fonte: Moraes (2012).

O modelo em questão abordou um pouco de área envolvendo biologia, psicologia, neuroanatomia, neurofisiologia, matemática e computação. Eliasmith não ficou surpreso ao concluir que o modelo conseguiu realizar as atividades propostas, no entanto admirou-se que os erros cometidos pelo modelo foram os mesmos erros dos seres humanos.

- **Psicologia:**

O método científico de Hermann von Helmholtz (1821-1894) em conjunto com seu aluno Wilhelm Wund (1821-1894) aplicaram-se ao estudo da visão humana é descrito até hoje como “o mais importante tratado sobre a física e a fisiologia da visão humana”, segundo Nalwa (1993), onde suas linhas que pesquisas se davam sobre o processos do pensamento. Enquanto os físicos estudavam uma forma de compreender a questão do pensamento dos seres humanos, os biólogos estudavam o comportamento animal. O behaviorismo surgiu como um movimento visto que os comportamentos animais careciam de dados introspectivos, desta forma estudavam os estímulos dos animais e suas respostas.

Uma característica importante da psicologia cognitiva para a IA é o fato de o cérebro ser considerado um dispositivo de processamento de informações. Os estudos de Kenneth Craik (1943), e logo depois Donald Broadbent (1958), se davam através de modelos de processamento de informações de fenômenos psicológicos. Já nos Estados Unidos, surgiu a ciência cognitiva como campo de desenvolvimento da modelagem de computadores, o surgimento se deu em um seminário, onde alguns documentos influenciaram nos modelos de computadores poderiam ser usados para tratar a psicologia da memória, a linguagem e o pensamento lógico. Segundo Anderson (1980), “uma teoria cognitiva deve ser igual a um programa de computador”, onde deve descrever em detalhes o fluxo da informação por meio do qual alguma função cognitiva poderia ser implementada.

- **Engenharia de Computadores:**

O computador foi a ferramenta que a Inteligência Artificial utilizou como artefato para introdução de seus conceitos. Na Segunda Guerra Mundial, a máquina de Turing foi desenvolvida com finalidade decifrar as mensagens alemãs. Konrad Zuse (1941) criou o primeiro computador programável e a primeira linguagem de programação de alto nível, daí em diante inicia-se a relação entre engenharia de computadores e Inteligência Artificial. Anterior a este feito, existiam os dispositivos de cálculo, no qual a primeira máquina programável foi desenvolvida por Joseph Marie Jacquard (1752-1834), onde utilizava cartões perfurados para armazenar instruções. Logo depois, Charles Babbage (1792-1871) projetou a Máquina Diferencial que se destinava a calcular tabelas matemáticas, porém não concluiu a mesma. O protótipo de Babbage só foi finalizado em 1991, onde se mostrou funcional, onde incluía memória endereçável, programas armazenados e saltos condicionais, sendo o primeiro artefato de executar computação universal. Ada Lovelace foi considerada a primeira programadora do mundo, utilizando a Máquina Analítica.

Segundo Russell (2004), a área de *software* de Ciência da Computação forneceu diversas ferramentas importantíssimas para a IA, tais como sistemas operacionais, as linguagens de programação e outras ferramentas necessárias para escrever programas modernos. Em contraproposta, as ideias de IA foram

aproveitadas na computação em geral incluindo o compartilhamento de tempo, interpretadores interativos, computadores pessoais com janelas e mouse, ambientes de desenvolvimento rápido, o tipo de dados lista encadeada, o gerenciamento automático de armazenamento e conceitos fundamentais de programação simbólica, funcional, dinâmica e orientada a objetos.

- **Teoria de controle e cibernética:**

A primeira máquina autocontrolada foi um relógio regulado com base no fluxo de água construído por Ctesibio de Alexandria (250 a.C). Depois desta invenção, surgiu o regulador de máquinas a vapor de James Watt (1736-1819), e o termostato de Cornelis Drebbel (1572-1633) que também inventou o submarino. No século XIX, foi desenvolvida a teoria matemática de sistemas realimentados estáveis.

A figura central na criação daquilo conhecido hoje como teoria de controle foi Nobert Wiener (1894-1964), um matemático que despertou interesse por sistemas por sistemas de controle biológico e mecânico e sua relação com a cognição. Junto de seus colegas, organizaram uma série de estudos onde exploravam novos modelos matemáticos e computacionais da cognição e influenciaram muitos outros pesquisadores na ciência do comportamento. O livro lançado por Wiener, intitulado *Cybernetics* (1948), abriu novos horizontes para possibilidade de máquinas dotadas de inteligência artificial.

Segundo Russell (2004), a moderna teoria de controle tem como objetivo o projeto de sistemas que maximizam a função objetivo sobre o tempo. Em tese, a Inteligência Artificial foi criada como forma de escapar das limitações da matemática e da teoria de controle na década de 1950.

- **Linguística:**

Para Santana (2013), a teoria do behaviorismo é um conceito generalizado que engloba as mais paradoxais teorias sobre o comportamento, dentro da psicologia. Em 1957, Skinner publica uma obra completa e detalhada da abordagem behaviorista para o aprendizado da linguagem, e logo depois Noam Chomsky publicou uma resenha onde mostrava que a teoria anteriormente dita não tratava a noção da criatividade da linguagem. A teoria

intitulada Syntactic Structures (Estruturas Sintáticas), explica o fato ocorrido e diferente de outras teorias, era formal o bastante para poder, em princípio, ser programada.

Partindo desta situação, a IA e a linguística surgiram na mesma época e se desenvolveram juntas, denominando num campo chamado linguística computacional ou processamento de linguagem natural, que consiste num método de interação entre homem máquina onde a informação seja de fácil entendimento para ambas as situações.

3.2 MODELOS DE INTELIGÊNCIA ARTIFICIAL

De acordo com Ganascia (1993), os principais modelos de Inteligência Artificial são:

- **Algoritmo Genético:** é um modelo para aprendizado de máquina. Segundo Charles Darwin, “somente os mais aptos sobrevivem”. Esta teoria evolucionista caracteriza os Algoritmos Evolutivos, que inclui um estudo sobre os Algoritmos Genéticos, estratégia de evolução, programação evolutiva e sistemas classificatórios. Criado por Holland (1975), os algoritmos genéticos tem como objetivo emular os operadores genéticos, tais como pelo cruzamento, mutação e reprodução. Desta forma, cria-se dentro da máquina uma população de indivíduos representados por cromossomos, onde estes indivíduos passam por um processo de simulado de evolução, seleção e reprodução, gerando uma nova população.
- **Programação Evolutiva:** semelhante ao algoritmo genético, porém dá maior ênfase na relação comportamental entre os parentes e descendentes. As soluções para os problemas são obtidas por meio de tentativas e transmitidas para a nossa população.
- **Lógica Fuzzy:** é uma metodologia que serve para representar, manipular e modelar informações incertas.
- **Sistemas baseados em Regras:** são sistemas que implementam comportamentos inteligentes de especialidades humanas.
- **Programação Genética:** é um campo voltado para a construção de programas que visam imitar o processo natural da genética, trabalhando com métodos de busca aleatória.

- **Raciocínio baseado em casos:** é um campo que utiliza uma grande biblioteca de casos para consulta e resolução de problemas. Os problemas atuais são resolvidos com base na recuperação e consulta de casos já solucionados anteriormente e da consequente adaptação das soluções encontradas.
- **Redes Neurais:** possui várias denominações, entre elas: Redes Neurais, Modelo Conexionista, Neurocomputação, Modelo de Processamento Paralelo Distribuído, Sistemas Neurofórmicos e Computadores Biológicos. Consistem em um número de elementos interconectados, conhecidos Neurônios, organizados em camadas que aprendem pela modificação da conexão firmemente conectando as camadas.

3.3 INTELIGÊNCIA E CONHECIMENTO

Segundo Rich (1993), para chegar à conclusão que uma ação tomada foi inteligente é necessário analisar todos os aspectos relativos à aquisição e desenvolvimento da inteligência. Desta maneira, o conhecimento é que faz possível o encadeamento e o desenvolvimento da inteligência.

Seguindo esta linha de raciocínio, surgiu o interesse de incorporar estes conhecimentos a sistemas computacionais de Inteligência Artificial. Desta maneira, há algumas características do conhecimento que devem ser analisadas:

- **Volume:** o conhecimento é muito volumoso, possui diversos aspectos ligados a si. À medida que é mais requisitado, mais informações são colocadas em questão.
- **Difícil caracterização:** não possui consciência do conhecimento que possui. Possui o conhecimento de certa forma, mas não sabe explicar como o adquiriu ou não sabe explicá-lo.

- **Conhecimento em constante mudança:** o conhecimento está crescendo passando por mudanças, crescendo, modificando e aperfeiçoando.
- **Diferente de dados (existe organização):** os dados fazem parte do conhecimento, mas o conhecimento em si não é um dado, desta forma os dados possuem certa lógica a modo de permitir sua interpretação.
- **Individual:** o conhecimento é individual. Duas pessoas possuem o mesmo conhecimento genérico, mas não de forma exata e idêntico. Cada um faz sua interpretação do conhecimento.

Computacionalmente, há necessidade de realizar uma estruturação do conhecimento a fim de verificar o que será utilizado, utilizando uma representação do conhecimento. Rich (1993) destaca que a representação do conhecimento possui as seguintes características:

- **Generalizável:** diferente do conhecimento que é individual, uma representação necessita de vários pontos de vista do mesmo conhecimento, de modo de possa ser atribuído a diversas situações e interpretações.
- **Conhecimento utilizável:** uma representação do conhecimento deve ser robusta, mesmo que incompleta e imprecisa, isto é, permite sua utilização mesmo que não satisfaça todas as questões possíveis.
- **Representação passível de atualização/ correção:** o conhecimento não é estático, desta maneira pode sofrer qualquer tipo de alteração e/ou correção.
- **Compreensível:** a representação deve ser compreensível ao ser humano a fim de permitir sua interpretação.

3.3.1 REPRESENTAÇÃO DO CONHECIMENTO

Segundo Fernandes (2003), a representação do conhecimento pode ser definida como um conjunto de convenções sintáticas e semânticas que tornam possível descrever coisas, onde se encontram duas diferentes entidades: fatos (coisas que se querem representar) e a representação dos fatos (formalismo utilizado como forma de manipular as informações).

Dentre as características da representação do conhecimento podemos destacar: detalhamento, alternativas de modelagem, clusterização do conhecimento, conhecimento explícito e flexível.

Segundo Fernandes (2003), existem diversos paradigmas de representação do conhecimento que têm emergido destas perspectivas:

- **Conhecimento procedural:** o conhecimento é representado através de procedimento ou funções;
- **Redes:** o conhecimento é representado através de um grafo direcionado, onde cada vértice representa representam os conceitos e objetos, as arestas são representadas pelas relações.
- **Frames:** segue o conceito do tópico anterior, porem cada nó representa conceitos e entidades, onde os mesmos possuem propriedades que podem ser especificadas ou herdadas.
- **Lógica:** um método de representar o conhecimento.
- **Árvores de decisão:** conceitos são organizados em forma de árvores.
- **Conhecimento estatístico:** faz uso de fatores de certeza, abordando diversas teorias.
- **Regras:** sistema que codifica as regras de ação/ condição.

- **Processamento paralelo distribuído:** utiliza de modelos conexionistas.
- **Esquemas híbridos:** combinação de técnicas que pode levar a uma técnica mais robusta e eficiente.
- **Casos:** com base nos conhecimentos adquiridos em casos anteriores, de forma analógica, encontra soluções para os casos enfrentados.

3.4 MÉTODOS DE BUSCA E REPRESENTAÇÃO DO CONHECIMENTO

Em conformidade Rich (1993), uma técnica de Inteligência Artificial é um método que explora o conhecimento e deve ser representado de tal forma que o conhecimento: capture generalizações, seja compreendido pelas pessoas que o fornecem, seja facilmente modificado para corrigir erros e refletir mudanças do mundo e da visão do mundo que o usuário possui, seja usado em inúmeras situações e que sirva de ajuda para superar seu próprio volume.

Dentre as técnicas de Inteligência Artificial tem a busca heurística, onde a mesma proporciona um meio de resolver problemas complexos. A palavra heurística vem do grego *heiriskein*, que significa “descobrir”. A heurística é um procedimento para resolver problemas através de um enfoque intuitivo, em geral racional, onde uma vez que identificado à estrutura do problema e explorando-a a fim de obter uma solução razoável. Para alguns estudiosos, as heurísticas são critérios, métodos ou princípios para decidir, entre várias alternativas quais são mais objetivas para atingir determinado objetivo.

As técnicas básicas de busca heurística são:

- **Busca em profundidade:** também conhecida como pilha, a busca em profundidade sempre expande o nó mais profundo na borda atual da árvore de busca. A busca prossegue imediatamente até o nível mais profundo da árvore de busca, onde os nós não têm sucessores. À medida que esses nós são expandidos, eles são retirados da borda, e então a busca “retorna” ao nó seguinte mais raso que ainda tem sucessores inexplorados.

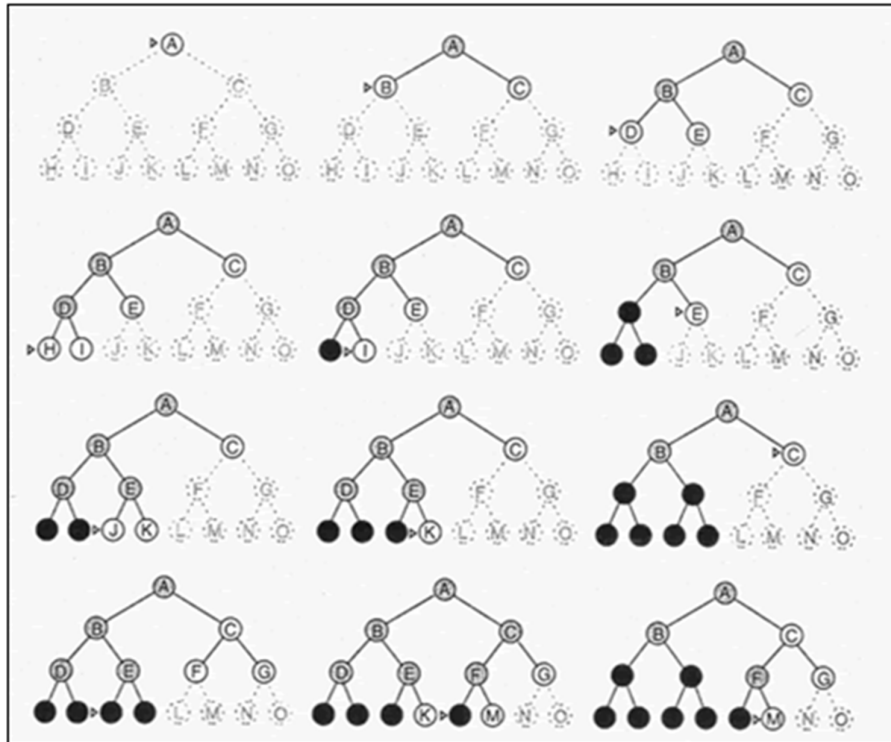


Figura 3 - Busca em profundidade em uma árvore binária
 Fonte: Russell e Norvig (2004)

- **Busca em largura:** também conhecido como “busca em amplitude”. Todos os nós de certo nível da árvore são examinados antes do nível abaixo, caso exista uma solução, e se o grafo é finito, a solução será encontrada. Esta técnica possui alguns inconvenientes, caso o grafo seja complexo haverá grande consumo memória exigindo um esforço computacional relativamente alto.

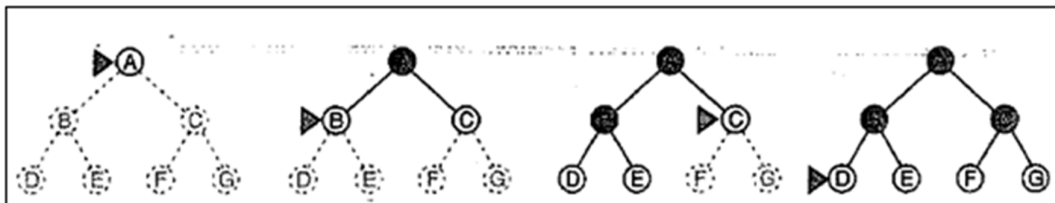


Figura 4 - Busca em extensão em uma árvore binária simples
 Fonte: Russell e Norvig (2004)

- **Gera e Testa:** é um procedimento de busca em profundidade, pois as soluções devem estar completas para serem testadas, onde a forma mais adequada de implementar o método é usar a busca em

profundidade adicionando *backtracking* (retrocesso com o objetivo de tentar uma outra alternativa de solução). Aceitável para solução de problemas simples ou como forma de auxiliar outras técnicas.

- **Busca em feixe:** é uma alternativa do método de busca em largura onde existe uma restrição ao número de nós que serão considerados em cada nível. É uma forma de reduzir o esforço de busca e encontrar bons resultados quando houver bons critérios de escolhas dos nós.
- **Subindo o morro:** a ideia deste método é maximizar a nota fornecida pela função heurística. A busca é feita com a geração de um estado sucessor do estado corrente. Se o estado sucessor tiver uma nota melhor, então ele passa a ser o estado corrente.

3.5 RESTRIÇÕES NA RESOLUÇÃO DE PROBLEMAS

Uma restrição pode ser considerada uma condição num vasto espaço de possibilidades. A representação matemática auxilia em todo um contexto onde fica mais claro e preciso cada informação, desta forma é possível restringir algumas variáveis de receber alguns valores visto que possuem alguma restrição.

O método utilizado para formular a resolução de um problema consiste em modelar todo o processo, a modo que fique claro todas as variáveis, o domínio de cada uma delas e o conjunto de restrições, de modo que todas estejam relacionadas entre si.

Segundo Quinet (2013), as restrições podem ser classificadas em:

- **Unárias:** restringe o valor em uma única variável, respeitando qualquer valor que não viole as restrições impostas;
- **Binárias:** relacionam duas variáveis;
- **De alta ordem:** quando envolvem três ou mais variáveis, os problemas criptoaritméticos é um exemplo;
- **De preferência:** quando certas atribuições tem prioridade sobre outras.

3.5.1 PROBLEMA DE SATISFAÇÃO DE RESTRIÇÕES

Um problema de satisfação de restrição possui esta definição por possuir um conjunto $X = \{x_1, x_2, \dots, x_n\}$ de variáveis, cada uma com seu domínio $D = \{d_1, d_2, \dots, d_n\}$ de valores possíveis e um conjunto $R = \{r_1, r_2, \dots, r_n\}$ de restrições que auxiliam nos valores que cada variável pode assumir.

Segundo Russell et al. (2004), um problema de satisfação de restrição (PSR) é definido por um conjunto de variáveis e restrições, onde cada variável possui um domínio não vazio de valores possíveis. Cada restrição envolve um subconjunto das variáveis especificando as combinações de valores permitidas para aquele subconjunto. Uma atribuição que não viola nenhuma restrição é considerada consistente ou válida, uma atribuição completa é aquela em que toda variável é mencionada. Analisando todo este contexto, a solução para uma PSR é uma atribuição completa que satisfaça todas as restrições.

A forma mais simples de visualizar um problema de satisfação de restrição é visualizá-lo como um grafo de restrições, conforme mostrado na Figura 5.

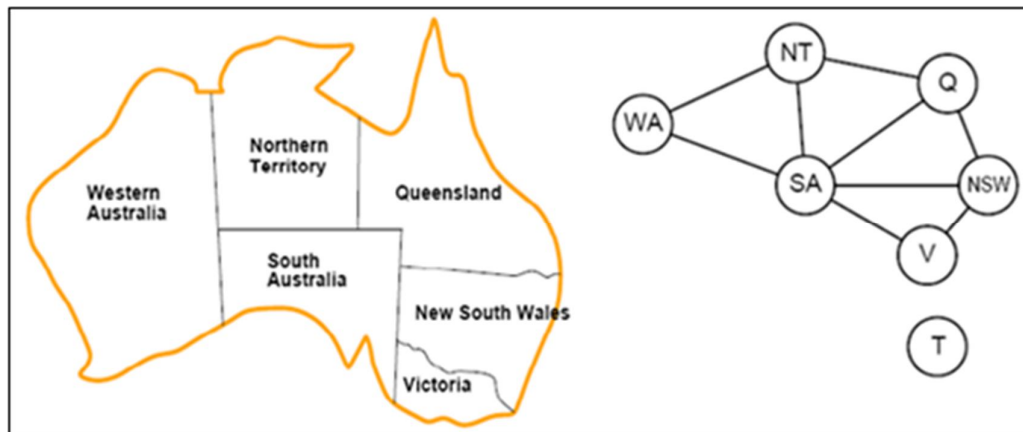


Figura 5 - Estados e territórios da Austrália.
Fonte: QUINET (2013)

A Figura 5 representa uma situação cujo objetivo é colorir cada região de forma que sua fronteira não possua a mesma cor que uma região adjacente.

No grafo, os nós correspondem às variáveis e os arcos correspondem às restrições.

O PSR pode receber uma formulação incremental como um problema de busca padrão da seguinte forma:

- **Estado inicial:** inicialmente a atribuição fica vazia;
- **Função sucessor:** os valores a serem inseridos nesta etapa, não podem ser conflitantes com os valores já preenchidos anteriormente;
- **Teste de objetivo:** a atribuição corrente é completa;
- **Custo de caminho:** custo constante para todo passo.

Analisando o contexto da coloração dos mapas citado anteriormente, é possível chegar a seguinte solução:

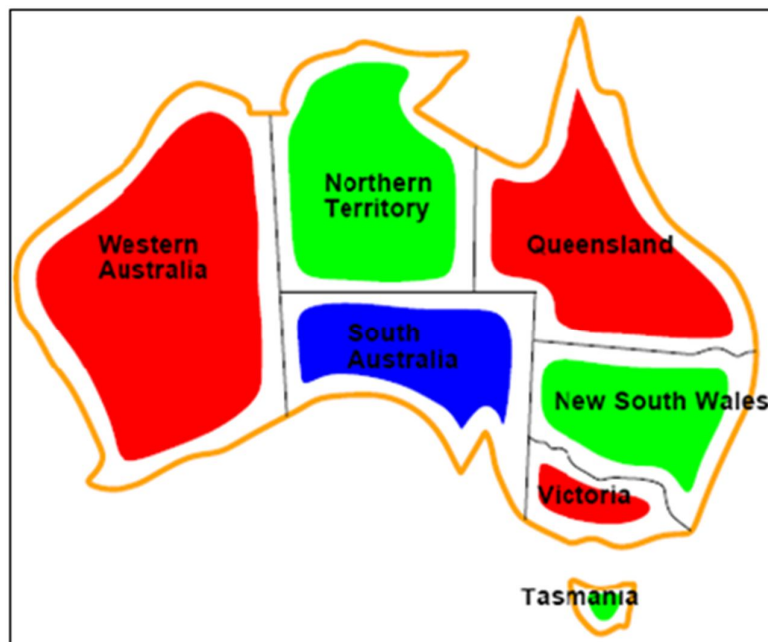


Figura 6 - Solução para o problema de coloração dos mapas.
Fonte: QUINET (2013)

Segundo Russell (2004), a busca com retrocesso é utilizada para indicar uma busca em profundidade que escolhe valores para uma variável de cada vez e que efetua o retrocesso quando uma variável não valores validos

restantes a serem atribuídos. Desta forma o retrocesso ocorre quando todos os valores são testados não alcançando o objetivo pré-estabelecido, assim a solução é retornar a variável anterior para alteração dos valores mediante as outras opções dentro do conjunto, ou seja, trata-se de um algoritmo recursivo. Devido ao grande consumo de tempo, é mais apropriado utilizar em problemas de pequeno porte.

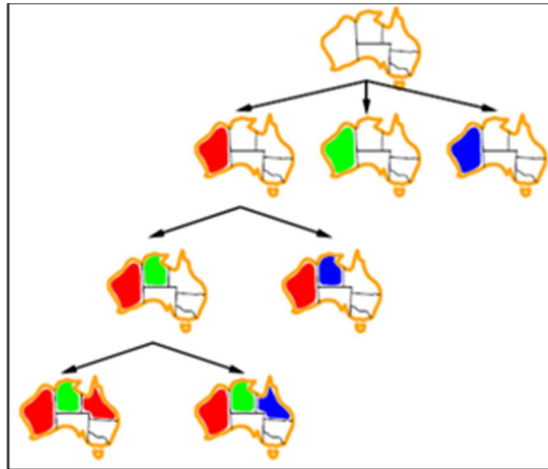


Figura 7 - Parte da árvore de busca utilizando o retrocesso simples.
Fonte: QUINET (2013)

A heurística de valores restantes mínimos (VRM) tem como ideia o processo de escolha da variável com menor número de valores válidos.

Segundo QUINET (2013), a heurística de grau tenta reduzir o fator de ramificação em escolhas futuras selecionando a variável envolvida no maior número de restrições sobre outras variáveis não atribuídas. Na Figura 68, o estado *South Australia* é a variável com grau mais alto, sendo cinco no caso. Seguindo esta ideia, é possível chegar a uma solução sem o retrocesso.

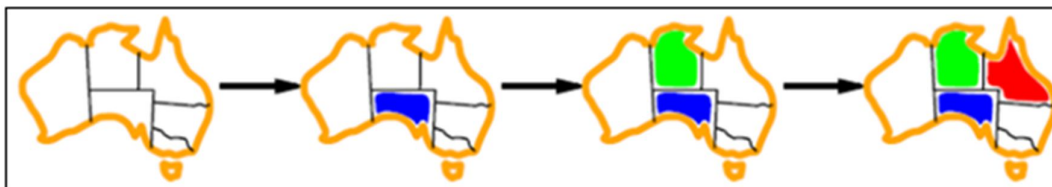


Figura 8 - A coloração azul discrimina o estado com maior grau.
Fonte: QUINET (2013)

A verificação prévia é um método utilizado para melhorar as restrições durante o processo de busca. No caso da coloração dos mapas, quando uma determinada região recebe um tipo de cor, imediatamente os territórios que fazem fronteiras não possuirão a mesma cor no seu conjunto, desta forma o processamento em busca da solução se torna mais ágil visto que as restrições são estreitamente manipuladas neste procedimento.

4 PROBLEMA DE GERAÇÃO DE HORÁRIO ESCOLAR NA UNIVERSIDADE SAGRADO CORAÇÃO

A Universidade Sagrado Coração é uma instituição de ensino superior localizada na cidade de Bauru, no estado de São Paulo. Atualmente a Universidade dispõe de aproximadamente 270 professores distribuídos nos 16 cursos na área de Exatas, 9 cursos na área de Saúde, 18 cursos na área de Humanas e 7 cursos de curta duração. Diante destes dados, é possível ter uma dimensão da quantidade de disciplinas vinculadas a cada curso e o nível de complexidade para gerar um horário.

Segundo Lobo (2005), um problema de quadro de horários (*timetabling problem*) consiste em fixar encontros entre professores e estudantes, dentro de horários pré-estabelecidos, obedecendo a determinadas restrições que mapeiam diversas políticas de cada instituição envolvida.

Semestralmente, os horários são gerados mediante a uma série de restrições envolvendo diversas vertentes, elevando assim o seu nível de complexidade. Para que um horário seja considerado ideal, ele deve atender as seguintes restrições:

- Preencher as disciplinas que fazem parte da grade do semestre;
- Preencher as disciplinas de acordo com a disponibilidade dos professores;
- Formular o horário sem lacunas;
- Priorizar o horário de segunda à sexta (período noturno, das 19h00min às 22h15min), caso não seja possível utilizar o sábado de manhã (das 08h00min às 11h15min);

No contexto desta pesquisa, foi analisado o curso de Ciência da Computação no período noturno, em que cada disciplina tem cerca de 1 a 4 créditos que corresponde à 1 hora-aula¹.

As possíveis restrições que fazem parte deste contexto são:

- Cada professor pode lecionar até quatro créditos por período;
- Cada professor pode lecionar mais de uma disciplina por semestre;
- Uma disciplina pode conter de um a quatro créditos por período;
- Disciplinas diferentes ministradas pelo mesmo professor não podem coincidir, em termos de horário;
- Um período deve ser preenchido, preferencialmente, por quatro créditos;
- As disciplinas de TCC (Trabalho de Conclusão de Curso) não precisam estar necessariamente no período noturno, portanto poderá estar em turno alternado;
- As disciplinas cadastradas como 'FIXAS', fazem referências àquelas disciplinas que demandam uma maior quantidade de alunos, sendo assim podem agrupar mais de uma turma;

Este conjunto de restrições representa o problema encontrado semestralmente pela Instituição de ensino, em que, dependendo da disponibilidade de cada professor, o processo tende a ficar mais complexo, diminuindo a quantidade de possibilidades para alcançar o objetivo.

¹ Uma hora-aula tem 60 minutos dos quais 45 minutos são cumpridos pelo aluno em sala de aula e 15 minutos correspondem às atividades desenvolvidas pelo discente fora da universidade

5 METODOLOGIA

5.1 DEFINIÇÃO DO ALGORITMO DE GERAÇÃO DE HORÁRIOS

O primeiro passo do processo de desenvolvimento consistiu em definir como seria o fluxo de informações da ferramenta e o seu algoritmo de funcionamento. A Figura 9 mostra esse fluxo.

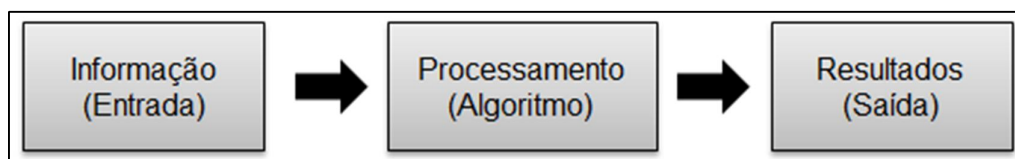


Figura 9 - Fluxo da Informação.
Fonte: Elaborada pelo Autor (2013).

Inicialmente as informações devem ser cadastradas pelo usuário, sendo elas relacionadas à Instituição, ao curso, a disciplina e relação curso-disciplina e aos professores, bem como suas restrições à disponibilidade e o vínculo professor-disciplina. O processamento tem como objetivo tratar a informação de entrada mediante o Algoritmo de Satisfação de Restrição (PSR), e por fim gerar o resultado esperado, respeitando a todas as condições impostas ao início do processo.

A heurística adotada no Algoritmo faz referência ao Professor. A formulação matemática desenvolvida tem como objetivo deixar o algoritmo mais eficiente e assertivo, onde são calculadas as informações pertinentes a cada professor da seguinte forma:

$$\text{Restritividade} = \left(\frac{\sum \text{disciplina}}{\text{Qtde. Disciplinas}} \times 100 \right) + \left(\frac{\sum \text{dias disponíveis}}{\text{Qtde. Dias}} \times 100 \right) / 2$$

Onde:

- \sum Disciplina – quantidade de disciplinas que Professor leciona dentro do horário a ser gerado;

- Qtde. Disciplinas – quantidade de disciplina envolvidas no horário a ser gerado;
- Σ dias disponíveis – disponibilidade do Professor mediante ao horário;
- Qtde. Dias – faz referência aos dias semana e esta variável por padrão possui um valor fixo ‘6’, contabilizado de segunda à sexta;
- Restritividade – é a média calculada entre o percentual de disponibilidade e o percentual das disciplinas que englobam o Professor;

Os dados de entrada são preenchidos mediante a uma consulta no Banco de Dados, onde são filtradas apenas as disciplinas que fazem referência ao Semestre do Curso em questão (considerando o semestre vigente), desta forma, o procedimento adotado para gerar o horário consiste na alocação dos dados do Professor em um objeto, onde o mesmo armazena as disciplinas em que o professor está vinculado bem como sua disponibilidade e seu percentual de Restritividade. Logo temos como exemplo:

Objeto I	
Professor	Patrick Pedreira
Disciplinas	Algoritmos (4) – Introdução a Computação (4)
Disponibilidade	Segunda – Quarta – Sexta
Restritividade	45%

Quadro 1 - Objeto exemplo com as informações do docente
Fonte: Elaborada pelo Autor (2013).

No Quadro 1, é possível analisar todas as informações que fazem referência ao professor Patrick, onde das 5 disciplinas relacionadas ao horário, o professor leciona 2 perfazendo 40% e dos 6 dias da semana (Segunda a Sábado) possui 3 dias perfazendo 50%, totalizando uma média de 45% de Restritividade. O valor “4” que acompanha a disciplina determina a quantidade de créditos.

Antes de realizar o Algoritmo de Satisfação de Restrição, todos os professores são alocados em objetos e organizados em lista mediante ao nível de restrição (Restritividade) de cada um, e se por ventura, acontecer do mesmo

professor possuir o mesmo nível de restrição, é analisado a questão da ordem alfabética por parte dos nomes de cada professor.

Analisando o contexto de uma forma geral, pode-se visualizar a seguinte estrutura com relação à geração dos horários:

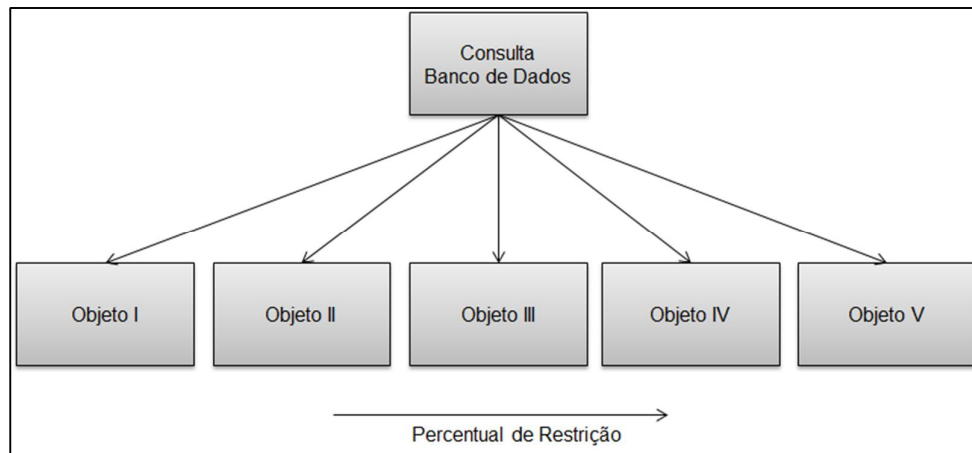


Figura 10 - Hierarquia para Geração do Horário.
Fonte: Elaborada pelo Autor (2013).

Na Figura 10 é possível visualizar a estrutura mencionada anteriormente, onde os objetos estão organizados em uma ordem atendendo ao percentual de restrição de cada Professor. A situação ilustra as informações que fazem referência ao 2º Semestre do Curso de Ciência da Computação, que abrange as seguintes disciplinas: Arquitetura de Computadores, Cálculo: Integração de uma variável, Comunicação e Expressão, Eletiva I e Técnicas de Programação. Uma vez que discriminado as disciplinas que compõem o Semestre é possível identificar os professores e sua disponibilidade respectivamente, conforme ilustra a Figura 11:

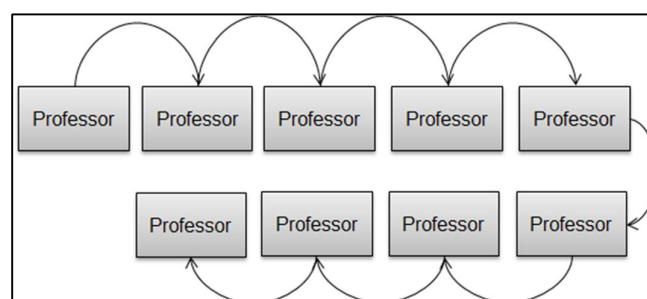


Figura 11 - Lista com a relação de Professores.
Fonte: Elaborada pelo Autor (2013).

Na Figura 11, é possível identificar a heurística adotada para iniciar o algoritmo, à medida que um Professor é alocado em um horário, é realizada a Verificação Prévia para que nenhum outro professor venha a preencher a mesma disciplina ou o mesmo dia da semana, ocasionando conflitos, se por ventura ao longo do processo o algoritmo não conseguir dar continuidade é realizado o Retrocesso, que consiste em um retorno no último Professor alocado e alterado seu dia da semana para outro dia. Com a utilização desta heurística, é possível minimizar a número de tentativas para gerar o horário. A seguir a mesma lista, porém com maiores informações referente às variáveis envolvidas:

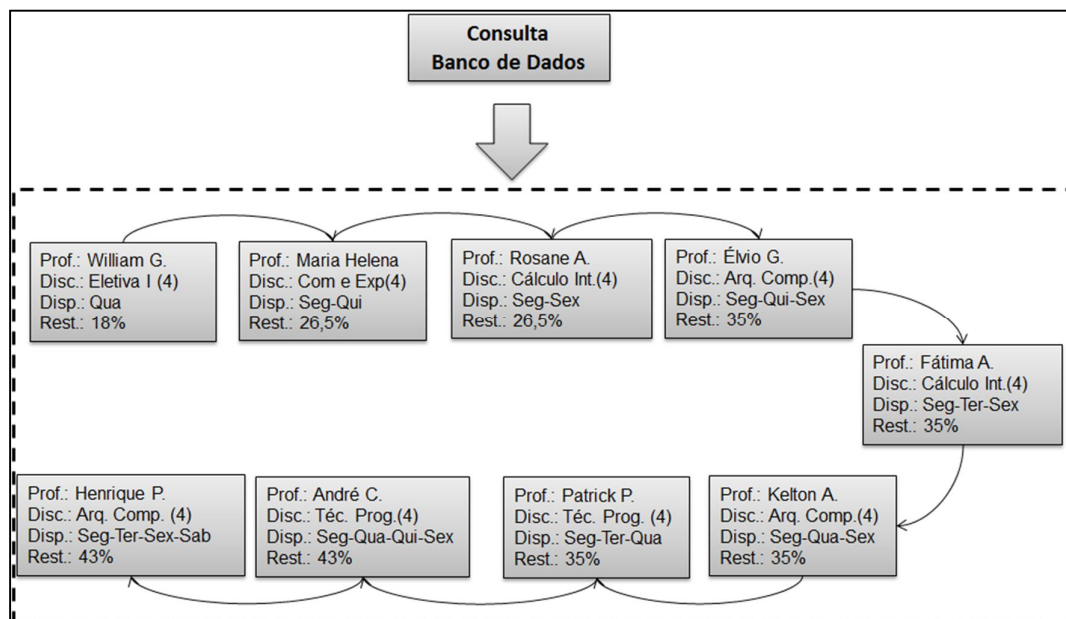


Figura 12 - Objetos carregados com as informações dos Professores.
Fonte: Elaborado pelo Autor (2013).

Com as informações abstraídas do Banco de Dados, é possível visualizar todos os objetos organizados de forma clara. A partir desta etapa inicia-se o Processamento do Algoritmo de Satisfação de Restrição, visto que os objetos possuem as informações pertinentes a cada professor. Desta forma, a cada alocação é realizada a Verificação Prévia (conforme descrito na seção 3.5.1) do seguinte modo:

- Quando uma disciplina é alocada juntamente com um professor, automaticamente nenhum outro professor lecionará a mesma disciplina;
- Quando uma disciplina é considerada “Fixa”, significa que abrange 2 turmas no mesmo horário, pois quando a mesma é preenchida ela replica as informações do 1º para o 2º horário;
- Quando um dia da semana é preenchido completamente, ou seja, alocou 4 créditos em um período automaticamente nenhuma outra disciplina poderá preencher o mesmo dia;
- Para controlar todas as alocações é realizada uma marcação em cada disciplina com o intuito de ser incrementada;

O Retrocesso ocorre quando o algoritmo chega ao final da lista e não alcança o resultado esperado, deste modo ele volta para a última alocação, e atribui o próximo dia da semana disponível para aquele professor, prosseguindo o algoritmo.

Na Figura 13, numa situação que exemplifica o processo de geração de horário, o professor William com o percentual de 18% de Restritividade tem alocada a quarta-feira com a disciplina Eletiva I de 4 créditos. Depois desta alocação, é realizada a verificação prévia nos demais objetos cujas informações são atualizadas a fim de que continue o processo.

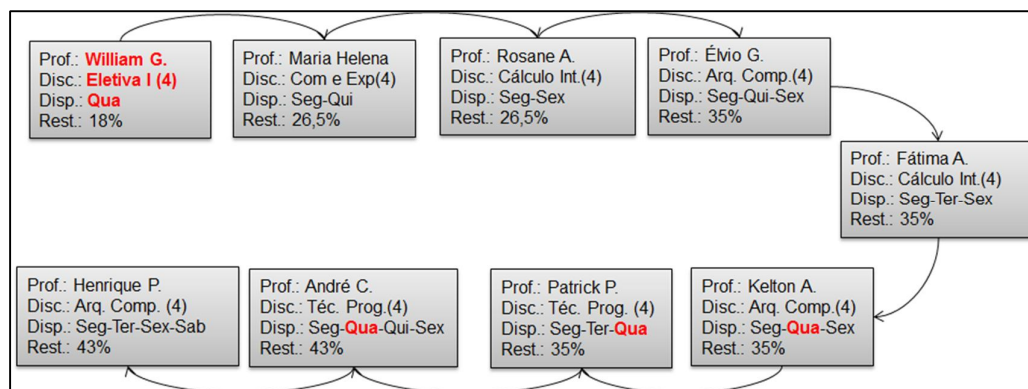


Figura 13 - Algoritmo [1].

Fonte: Elaborada pelo Autor (2013).

Na Figura 14, a professora Maria Helena tem alocada a segunda-feira com a disciplina de Comunicação e Expressão de 4 créditos.

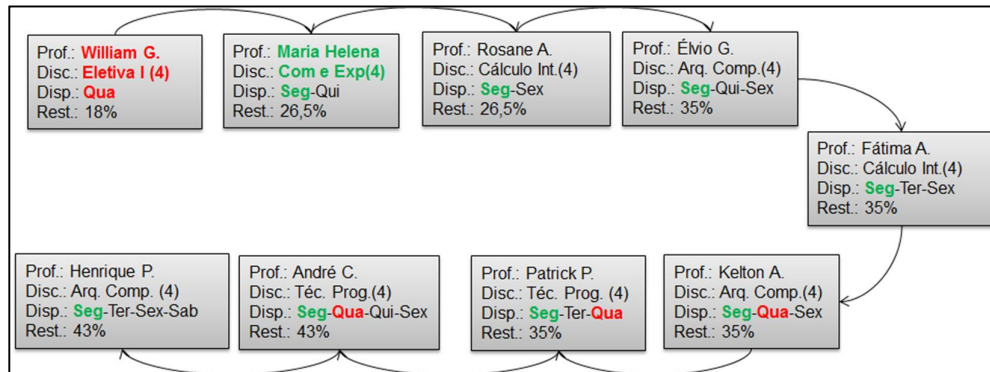


Figura 14 - Algoritmo [2]
Fonte: Elaborada pelo Autor (2013).

Na Figura 15, a professora Rosane tem alocada a sexta-feira com a disciplina de Cálculo: Integração de uma variável de 4 créditos.

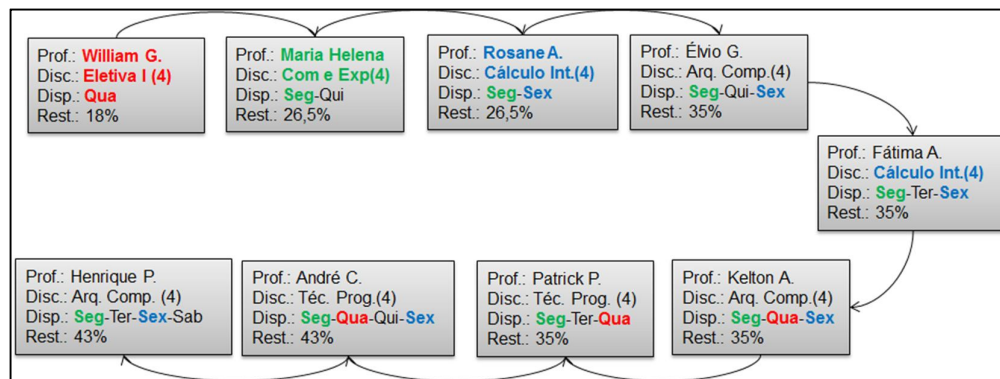


Figura 15 - Algoritmo [3].
Fonte: Elaborada pelo Autor (2013).

Na Figura 16, o professor Elvio tem alocada a quinta-feira com a disciplina de Arquitetura de Computadores de 4 créditos. Nota-se que os próximos professores, Fátima e Kelton, não possuem disciplinas a preencher dentro do horário ou não possuem dias disponíveis, pois os mesmos já foram alocados para outros professores, por fim o algoritmo procura o próximo professor disponível na lista para dar continuidade ao processo.

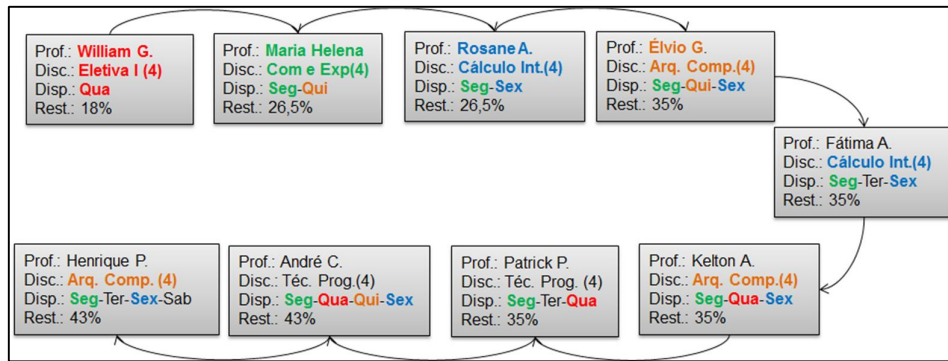


Figura 16 - Algoritmo [4]
Fonte: Elaborada pelo Autor (2013).

Na Figura 17, é possível visualizar que o algoritmo alcançou o resultado esperado. Neste exemplo ficou evidente que a verificação prévia exerce um papel fundamental na atualização das informações mediante cada locação. Quando por algum motivo o algoritmo não consegue prosseguir adiante é realizado o retrocesso, entretanto se o mesmo não contribuir, a heurística sofre uma alteração passando a ser inicializada pela variável menos restritiva, ou seja, realizando o processo inverso. Se por acaso o algoritmo não conseguir alcançar o resultado esperado, é gerada uma mensagem informando ao usuário que não foi possível gerar o horário.

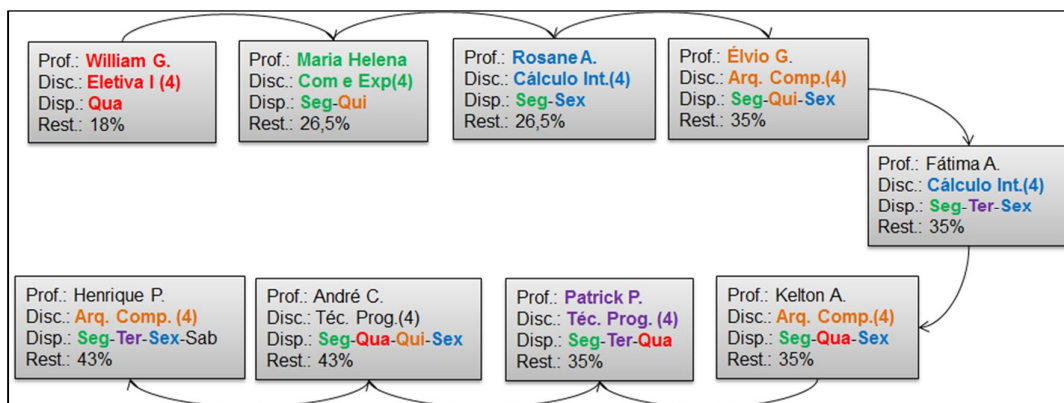


Figura 17 - Algoritmo [5].
Fonte: Elaborada pelo Autor (2013).

Na Figura 18, é possível visualizar os professores alocados dentro do horário final.

Segunda	Terça	Quarta	Quinta	Sexta	Sábado
Maria H. Com. Exp.	Patrick P. Téc. Prog.	William G. Eletiva I	Élvio G. Arq. Comp.	Rosane A. Cálculo	
Maria H. Com. Exp.	Patrick P. Téc. Prog.	William G. Eletiva I	Élvio G. Arq. Comp.	Rosane A. Cálculo	
Maria H. Com. Exp.	Patrick P. Téc. Prog.	William G. Eletiva I	Élvio G. Arq. Comp.	Rosane A. Cálculo	
Maria H. Com. Exp.	Patrick P. Téc. Prog.	William G. Eletiva I	Élvio G. Arq. Comp.	Rosane A. Cálculo	

Figura 18 - Horário Ciência da Computação 2º Semestre.
Fonte: Elaborada pelo Autor (2013).

As seções seguintes mostram o desenvolvimento do Banco de Dados e o desenvolvimento da aplicação.

5.2 DESENVOLVIMENTO

A modelagem do banco de dados foi desenvolvida com a utilização do Case Studio 2 devido à praticidade para realização da engenharia reversa. Para manipulação das informações no banco de dados foi utilizando a Microsoft SQL Server Management Studio 2012. A Figura 19 representa a modelagem Entidade Relacionamento:

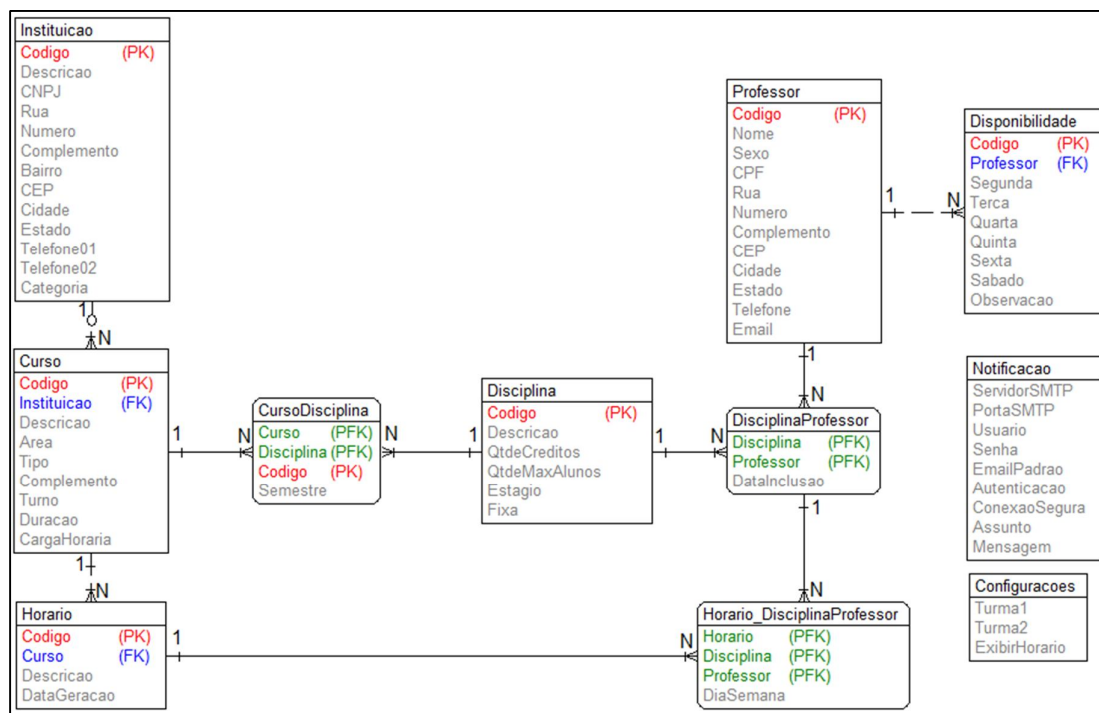


Figura 19 - Modelo Entidade Relacionamento (MER).
Fonte: Elaborada pelo Autor (2013).

Na Figura 19, a tabela Notificação está apartada, pois diz respeito à configuração de *e-mails* para envio. Entende-se que os valores '1' e 'N' dizem respeito à cardinalidade de cada tabela. A seguir, é relatada a finalidade de cada tabela dentro do modelo de Entidade Relacionamento apresentado:

- **Instituição:** nesta tabela estão alocadas todas as informações que referenciam a Instituição, neste caso a Universidade Sagrado Coração.
- **Curso:** nesta tabela estão alocadas todas as informações que referenciam o curso, partindo de uma única Instituição que pode abranger diversos cursos em diversas áreas relacionadas.
- **Disciplina:** nesta tabela estão alocadas todas as informações que referenciam as disciplinas. Considerando que uma mesma disciplina pode contemplar diversos cursos e que um curso pode ser contemplado por diversas disciplinas, se fez necessária à inserção de uma tabela intermediária entre Curso e Disciplina para maior controle dos dados.
- **Curso_Disciplina:** nesta tabela estão registradas todas as informações que referenciam a ligação entre Curso e Disciplina, como já mencionado anteriormente, o campo Semestre mostra a relação entre as tabelas;
- **Professor:** nesta tabela estão registradas todas as informações que referenciam um professor.
- **Disponibilidade:** nesta tabela estão registradas todas as informações que referenciam a disponibilidade de cada professor, onde a ligação se dá de 1→1. A tabela em questão possui todos os dias da semana em campos do tipo Bit (1 significa que o professor tem o dia disponível e 0 significa que está indisponível).
- **Disciplina_Professor:** nesta tabela estão registradas todas as informações que referenciam a relação entre Disciplina-Professor, visto

que uma disciplina pode haver mais um professor alocado e que um professor pode lecionar mais de uma disciplina, se fez necessária a inserção de uma tabela intermediária entre Disciplina e Professor para maior controle dos dados.

- **Horario:** nesta tabela estão registradas todas as informações que referenciam os horários gerados, sendo que um horário necessariamente precisa estar vinculado a um curso específico.
- **Horario_DisciplinaProfessor:** nesta tabela estão registradas todas as informações que referenciam os horários gerados, bem como os professores, disciplinas e o dia de semana.
- **Notificacao:** nesta tabela estão alocadas todas as informações que referenciam as configurações para envio de *e-mail*, onde é possível salvar apenas uma conta.
- **Configuracoes:** nesta tabela estão alocadas todas as informações que referenciam as configurações, onde fica discriminada a nomenclatura das Turmas e um campo Bit nomeado 'ExibirHorario' que quando ativado exibe o horário após gerado, caso contrário apenas salva as informações.

Respeitando o fluxo da informação apresentado na Figura 9, a estrutura da aplicação consiste em:

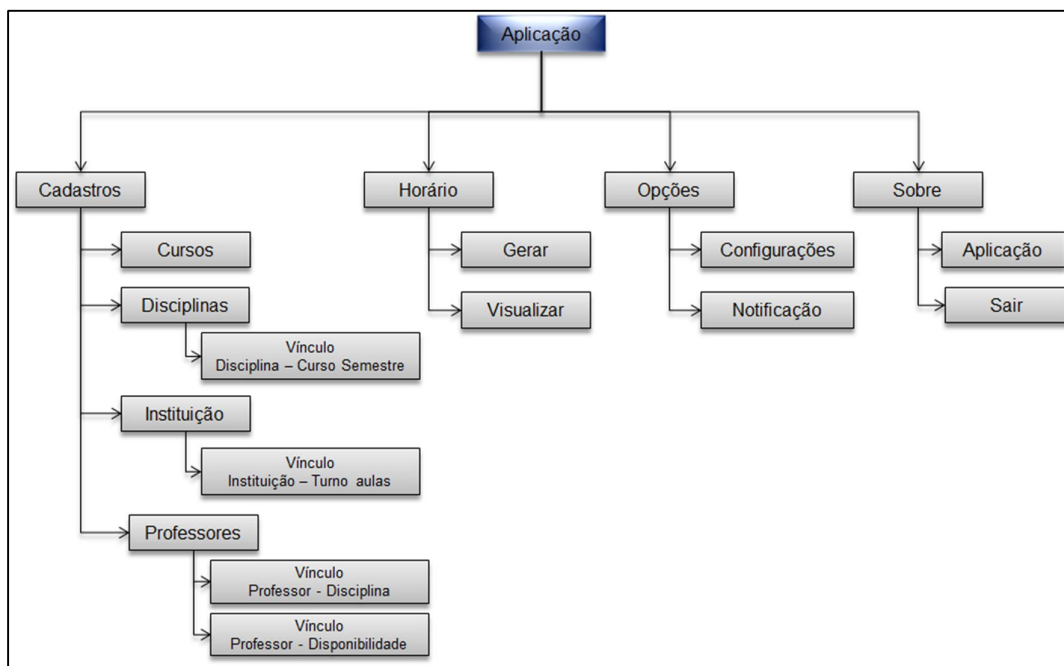


Figura 20 - Estrutura geral da aplicação.
Fonte: Elaborada pelo Autor (2013).

Na Figura 20 é possível identificar as funcionalidades do sistema descritas da seguinte forma:

- **Cadastro de Cursos:** manipulação das informações pertinentes ao Curso. Neste cadastro permite que os mesmos sejam cadastrados conforme oferecidos;
- **Cadastro de Disciplinas:** manipulação das informações pertinentes às disciplinas. Neste cadastro é possível vincular as disciplinas ao semestre de cada curso (1→N) conforme grade curricular;
- **Cadastro de Instituições:** manipulação das informações pertinentes à Instituição;
- **Cadastro de Professores:** manipulação das informações pertinentes aos professores. Neste cadastro é possível vincular os professores às disciplinas em que estão aptos a lecionar, e também informar a disponibilidade para o semestre vigente;

- **Gerar:** geração dos horários mediante ao preenchimento de alguns parâmetros. Nesta etapa o usuário pode optar por notificar os professores via e-mail sobre os horários gerados, apenas se tiver uma conta configurada;
- **Visualizar:** visualização das informações que contemplam os horários já gerados;
- **Configurações:** manipulação das configurações da aplicação;
- **Notificação:** manipulação das informações pertinentes à configuração do servidor de *e-mail*, nesta etapa é configurada a conta de *e-mail* que irá encaminhar aos professores os horários gerados;
- **Aplicação:** informações da aplicação;
- **Sair:** encerra aplicação.

Na tela principal (Figura 21) é possível visualizar todas as funcionalidades do sistema.

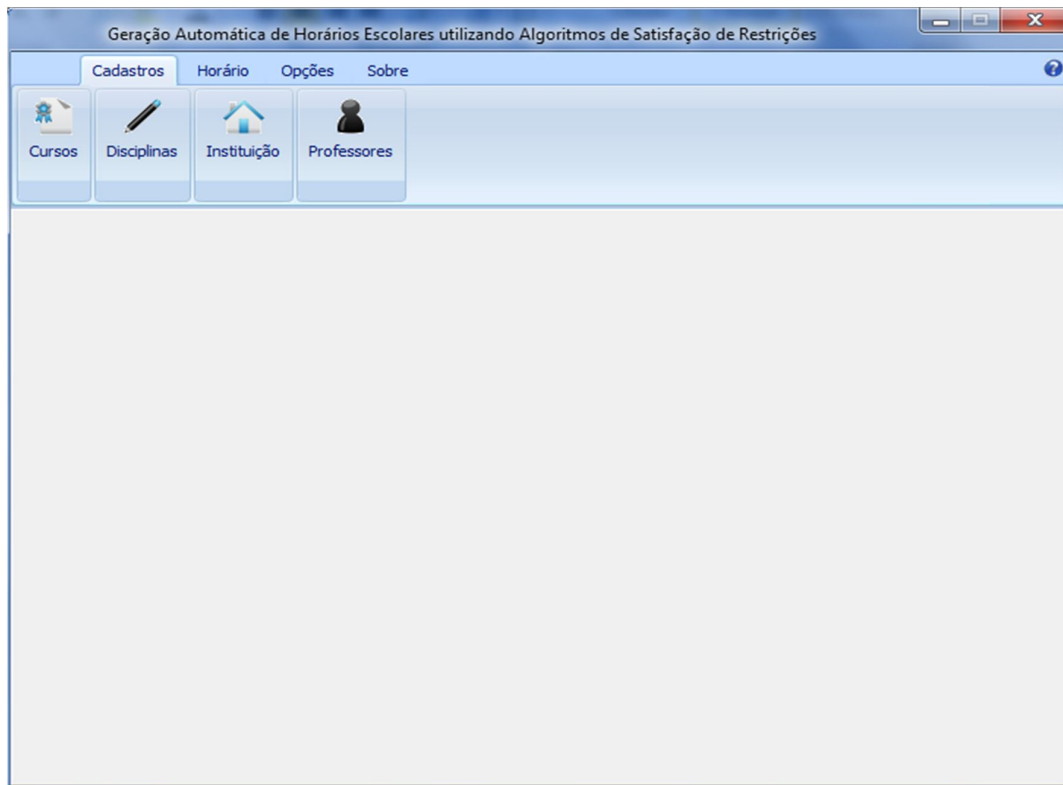


Figura 21 - Tela Inicial Aplicação.
Fonte: Elaborada pelo Autor (2013).

Foi desenvolvida uma classe modelo para cadastro (Pai) que consiste nas funcionalidades básicas de inserção, edição e exclusão, a fim de que as classes filhas pudessem herdar as devidas funcionalidades, informando apenas a tabela e os campos a serem manipulados.

No Cadastro de Cursos (Figura 22) é possível interagir com todas as informações que referenciam o mesmo, tais como a descrição do curso, a Instituição em que esta vinculada, a área (Biológicas/Exatas/Humanas), tipo de curso (Bacharelado/Licenciatura/Tecnólogo), a duração, o turno e a carga horária.

Código	Descrição	Tipo	Turno	Duração
001	Ciência da Computação	Graduação	Diurno	04 anos
002	Engenharia da Computação	Tecnólogo	Noturno	05 anos
003	Ciências Contábeis	Tecnólogo	Diurno	02 anos
004	Gastronomia	Graduação	Diurno	04 anos
005	Educação Física	Tecnólogo	Diurno	02 anos
006	Matemática	Graduação	Noturno	04 anos
007	Publicidade e Propaganda	Graduação	Noturno	04 anos

Informações Gerais

Código: 001

Descrição: Ciência da Computação

Área: Exatas | Tipo: Graduação

Turno: Diurno | Carga Horária: 3600

Instituição: 001 | Descrição: Universidade Sagrado Coração

Complemento: Lato Sensu | Duração: 04 anos

Figura 22 - Cadastro de Cursos.
Fonte: Elaborada pelo Autor (2013).

No Cadastro de Disciplinas (Figura 23) foi possível manipular todas as informações que referenciam o mesmo, tais como a descrição da disciplina, a quantidade de créditos e a quantidade de alunos. Este último campo é analisado de forma isolada para verificar se a quantidade de alunos ultrapassa o valor de 25, caso isto ocorra, ela é considerada como uma 'disciplina espelho' nesta situação, caso 2 horários sejam gerados ao mesmo tempo, esta disciplina vem a contemplar os 2 horários no mesmo dia.

Código	Descrição	Qtde Créditos	Qtde Alunos
001	Fundamentos de Administração	1 Créditos	40 Alunos
002	Algoritmos	4 Créditos	20 Alunos
003	Cálculo: Limite e Derivada	3 Créditos	50 Alunos
004	Introdução à Computação	4 Créditos	20 Alunos
005	Introdução à Lógica	4 Créditos	20 Alunos
006	Cálculo: Integração de uma variável	4 Créditos	30 Alunos
007	Comunicação e Expressão	4 Créditos	40 Alunos

Informações Gerais

Código: 001

Descrição: Fundamentos de Administração

Quantidade Créditos: 01

Quantidade Alunos: 40

Figura 23 - Cadastro de Disciplinas.
Fonte: Elaborada pelo Autor (2013).

Na segunda etapa do Cadastro de Cursos (Figura 24) é estabelecido o vínculo entre a disciplina e o semestre do curso. Visto que um curso pode abranger diversas grades, a disciplina pode estar vinculada em mais de um semestre para o mesmo curso.

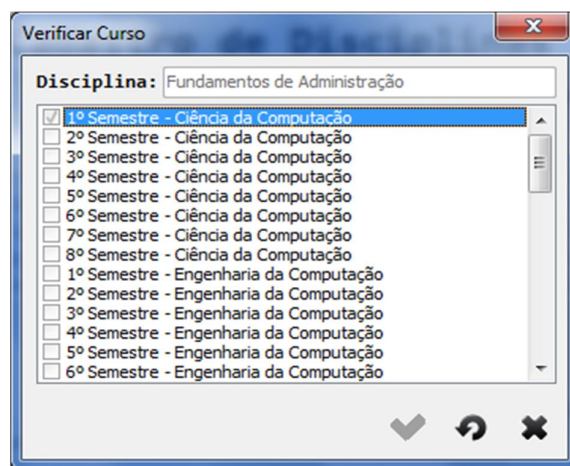


Figura 24 - Cadastro de Cursos – Semestre.
Fonte: Elaborada pelo Autor (2013).

No Cadastro de Instituições (Figura 25) é possível manipular todas as informações que referenciam a mesma, tais como nome, endereço e contatos telefônicos.

A janela 'Cadastro de Instituições' exibe uma tabela com uma única linha de dados e uma seção de formulário 'Informações Gerais'.

Código	Descrição	Cidade	Estado	Telefone01	Telefone02
001	Universidade Sagrado Coração	Bauru	SP	(14)02107-7000	(14)99999-9999

Informações Gerais

Código: 001

Descrição: Universidade Sagrado Coração

CNPJ: 11.111.111/1111-11

Categoria: Instituição Privada

Rua: Rua Irmã Arminda

Numero Complemento: 10 50

Bairro: Vl Cardia

CEP: 17016-170

Cidade: Bauru

Estado: SP

Telefone 01: (14)02107-7000

Telefone 02: (14)99999-9999

Figura 25 - Cadastro de Instituições.
Fonte: Elaborada pelo Autor (2013).

No cadastro de Professores (Figura 26) é possível manusear todas as informações que referenciam o nome, endereço, telefone e *e-mail*. Algumas validações foram criadas nesta etapa como forma de contribuir para processos futuros, como é o caso do e-mail. Este campo deve ser preenchido de forma correta, pois em uma futura notificação a mensagem não será entregue.

Código	Nome	Telefone	Email
001	Patrick Pedreira Silva	(14)99999-5555	patrick.silva@usc.br
002	Élvio Gilberto Silva	(14)03011-0000	egsilva@usc.br
003	Erica Paveloski	(14)09999-9999	erica.paveloski@usc.br
004	Rosane Araújo	(14)03011-0000	rosane.araujo@usc.br
005	Henrique Pachioni Martins	(14)99999-9999	henrique.martins@usc.br
006	Márcio Cardim	(14)98888-8888	marcio.cardim@usc.br
007	Fátima Araújo	(14)99999-9999	fatima.araujo@usc.br

Informações Gerais

Código: 001

Nome: Patrick Pedreira Silva CPF: 111.111.111-11 Sexo: Masculino

Rua: Rua Irmã Arminda Número: 10 Complemento: 85 CEP: 17000-000

Cidade: Bauru Estado: SP Telefone: (14)99999-5555 Email: patrick.silva@usc.br

Figura 26 - Cadastro de Professores.
Fonte: Elaborada pelo Autor (2013).

Na Figura 27, são listadas todas as disciplinas previamente cadastradas na aplicação, neste trecho é possível realizar o vínculo entre professor-disciplina, a fim de que o mesmo possa lecionar diversas disciplinas. Esta etapa se faz necessária, pois no ato da geração do horário, é este vínculo que será analisado.

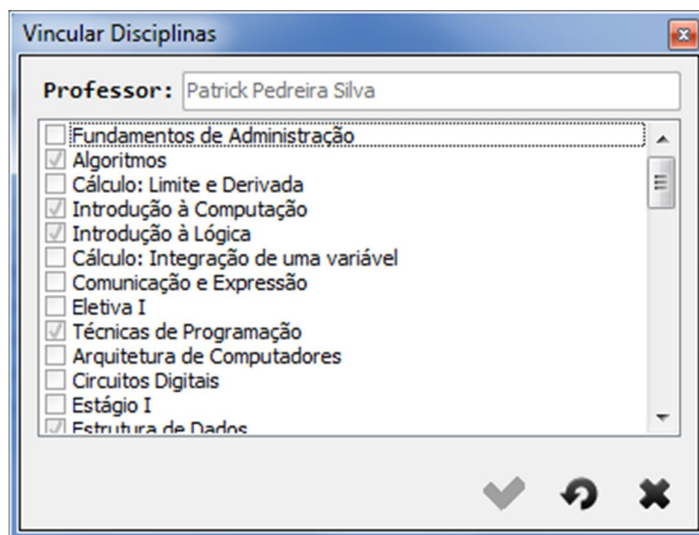


Figura 27 - Cadastro de Professores, vínculo com as disciplinas.
Fonte: Elaborada pelo Autor (2013).

Na etapa de cadastro de professores (Figura 28) é preenchida a disponibilidade de cada professor. Caso o professor seja vinculado a um horário, automaticamente sua disponibilidade deixará de existir.

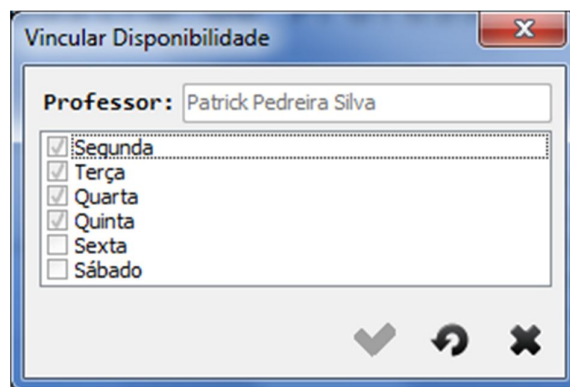


Figura 28 - Cadastro de Professores - Disponibilidade.
Fonte: Elaborada pelo Autor (2013).

A Figura 29 ilustra a tela de Geração de Horários. Nesta etapa o usuário deve preencher todos os parâmetros necessários. Quando o usuário optar pelo curso, automaticamente a quantidade de semestres é preenchida conforme a duração do curso informado no ato do cadastro (Figura 22). Após selecionar o semestre, ao lado em 'Selecione as disciplinas' são carregadas as disciplinas que estão vinculadas ao semestre do curso, conforme cadastro

(Figura 24), onde o usuário pode optar por quais disciplinas deseja gerar o horário. Por fim, em 'Notificar Professores', o usuário tem a praticidade de notificar os professores vinculados ao horário via *e-mail*.

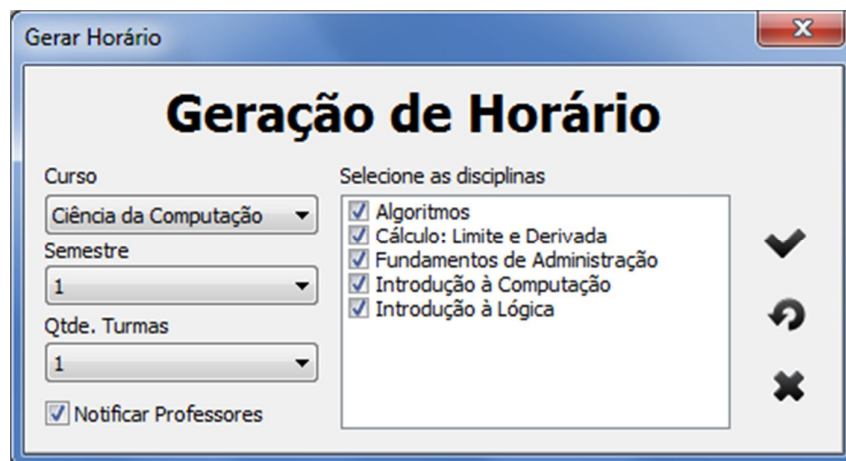


Figura 29 - Geração de Horário.
Fonte: Elaborada pelo Autor (2013).

Na Figura 30 é possível visualizar os horários gerados, com a possibilidade de visualizar maiores informações no botão 🔍 'Visualizar'.

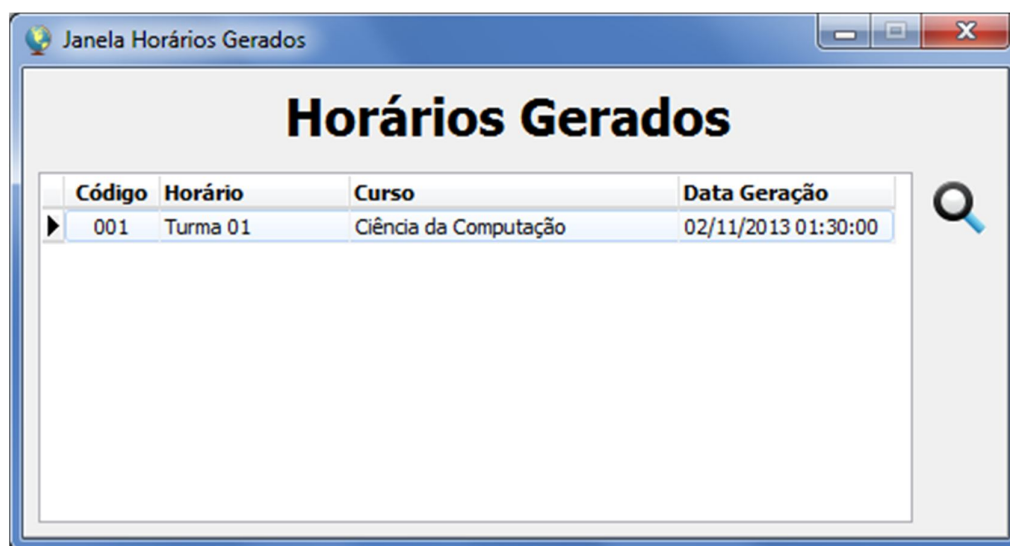



Figura 30 - Horários gerados.
Fonte: Elaborada pelo Autor (2013).

Na Figura 31 trata-se da Tela de Notificação, nesta etapa é preciso configurar todos os campos para o envio de *e-mail*, após o preenchimento basta clicar no botão  'Consultar Conexão' para verificar se a conta esta configurada corretamente. Foram utilizados alguns colchetes no campo "Assunto" e no campo "Mensagem" com o propósito de deixar as informações mais dinâmicas, no ato do envio onde está '[Professor]' ficará o nome do professor.

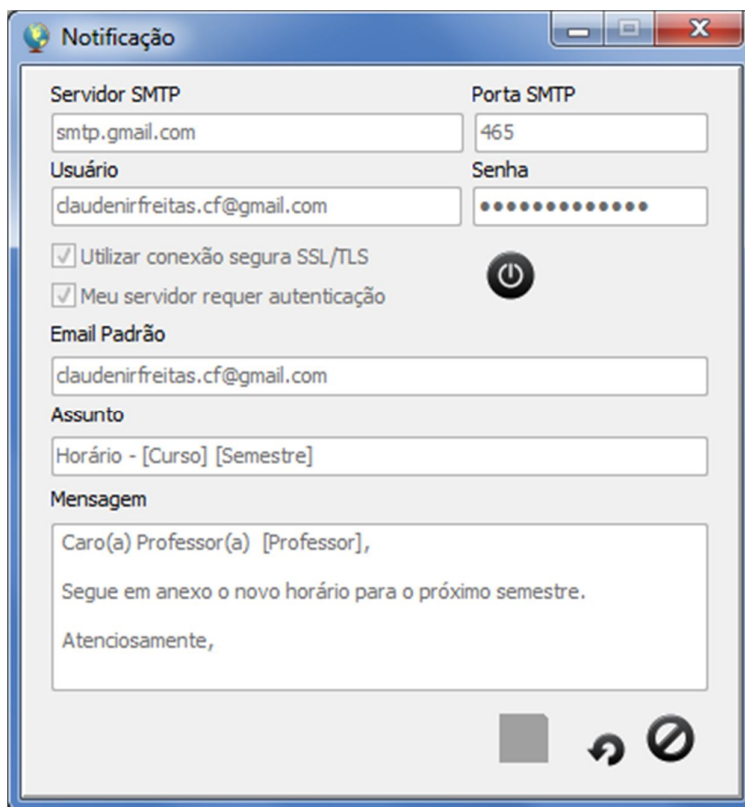


Figura 31 - Configurações de Notificação.
Fonte: Elaborada pelo Autor (2013).

A Figura 32 ilustra a Tela de Configurações. Nesta etapa é possível informar a nomenclatura das turmas, e se ao término da geração o horário será exibido ou não.

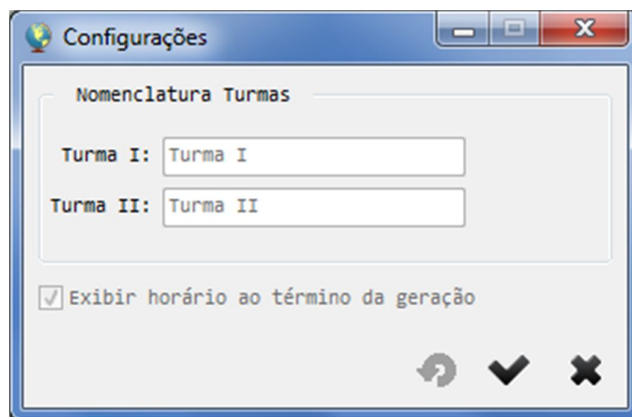


Figura 32 - Configurações da aplicação.
Fonte: Elaborada pelo Autor (2013).

A tela “Sobre” (Figura 33) traz algumas informações referentes ao trabalho em questão.

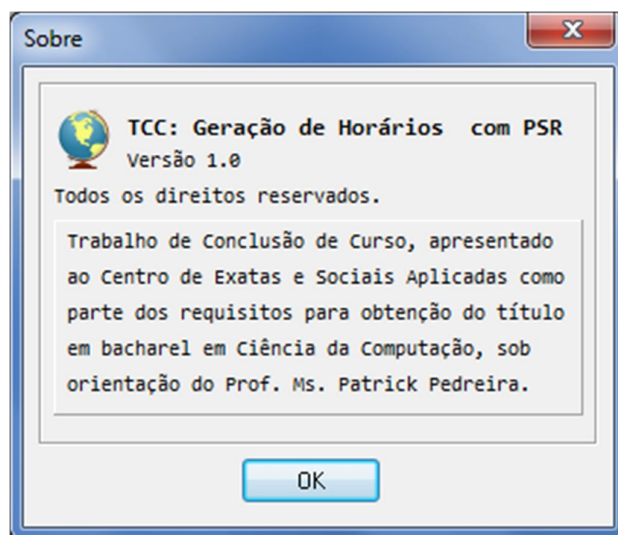


Figura 33 - Sobre. Fonte: Autor.
Fonte: Elaborada pelo Autor (2013).

6 AVALIAÇÃO DO SISTEMA

Este trabalho apresenta a utilização de algoritmos de satisfação de restrições na construção da grade horária para instituições de ensino superior. Procurou-se mostrar a complexidade dos problemas encontrados para definir uma solução de boa qualidade para este problema, uma vez que, existem múltiplas objetivas, múltiplas restrições e um número considerável de variáveis envolvidas.

Em virtude desta complexidade, considera-se que a aplicação de algoritmos de satisfação de restrições seja apropriada para utilização na automação do processo.

Apesar de haver restrições que sejam comuns a qualquer instituição de ensino, existem algumas restrições que são específicas de cada instituição e estas restrições podem dificultar o processo de implementação de um sistema de geração de grade horária de propósito geral.

Na presente seção relatam-se os resultados obtidos no processo de um experimento realizado considerando geração de uma grade horária para o curso de Ciência da Computação da Universidade Sagrado Coração, na cidade de Bauru - SP.

Para testar o protótipo foi utilizado o conjunto de dados apresentado na Figura 34:

Código	Professor	E-mail	S	T	Q	Q	S	S
1	Patrick Pedreira Silva	patrick.silva@usc.br	1	1	1	1	0	0
2	Élvio Gilberto Silva	egsilva@usc.br	1	1	0	1	1	0
3	Erica Paveloski	erica.paveloski@usc.br	1	0	0	0	1	0
4	Rosane Araújo	rosane.araujo@usc.br	1	1	1	0	1	0
5	Henrique Pachioni Martins	henrique.martins@usc.br	1	1	0	0	1	1
6	Márcio Cardim	marcio.cardim@usc.br	1	1	0	1	1	1
7	Fátima Araújo	fatima.araujo@usc.br	1	1	0	1	1	0
9	André Castro	andre.castro@usc.br	1	0	1	1	1	0
10	Anderson Talon	anderson.talon@usc.br	1	0	1	1	1	0
12	Márcia Marinho	marcia.marinho@usc.br	1	0	1	0	1	0
13	Kelton Augusto Pontara	kelton.pontara@usc.br	1	0	1	0	1	0
16	William Galvão	william.galvao@usc.br	1	1	0	1	1	0
17	Maria Helena	maria.helena@usc.br	1	0	0	1	1	0

Figura 34 - Relação Dados Professor.
Fonte: Elaborada pelo Autor (2013).

Na Figura 34 é possível visualizar todos os dados que dizem respeito aos professores, bem como o *e-mail* e a disponibilidade. Esses dados são necessários para demonstração dos exemplos a seguir.

Na Figura 35 estão discriminados todos os dados que referenciam o curso, bem como as disciplinas e suas especificações. A coluna 'Fixa' esta discriminada em um campo BIT, onde se porventura a quantidade de alunos ultrapassarem a quantidade de 25 alunos, significa que a disciplina é considerada uma disciplina espelho, ou seja, contempla 2 turmas com o mesmo horário, desta forma o Bit recebe 1, caso contrário recebe 0.

Curso	Semestre	Disciplina	Qtde. Créditos	Qtde. Alunos	Fixa
Ciência da Computação	1	Algoritmos	4	20	0
Ciência da Computação	1	Cálculo: Limite e Derivada	4	50	1
Ciência da Computação	1	Fundamentos de Administração	4	40	1
Ciência da Computação	1	Introdução à Computação	4	20	0
Ciência da Computação	1	Introdução à Lógica	4	20	0
Ciência da Computação	2	Arquitetura de Computadores	2	25	0
Ciência da Computação	2	Cálculo: Integração de uma variável	4	30	1
Ciência da Computação	2	Comunicação e Expressão	4	40	1
Ciência da Computação	2	Eletiva I	4	20	0
Ciência da Computação	2	Técnicas de Programação	4	20	0
Ciência da Computação	3	Circuitos Digitais	2	20	0
Ciência da Computação	3	Estatística	2	30	1
Ciência da Computação	3	Estrutura de Dados	4	20	0
Ciência da Computação	3	Fundamentos de Sistema de Informação	2	20	0
Ciência da Computação	3	Linguagem de Programação I	3	20	0
Ciência da Computação	3	Métodos e Técnicas de Pesquisa	4	50	1
Ciência da Computação	3	Pesq. de Prática em Ciência da Computação I	1	20	0
Ciência da Computação	4	Banco de Dados I	4	20	0
Ciência da Computação	4	Engenharia de Software I	2	20	0
Ciência da Computação	4	Estágio II	1	20	0
Ciência da Computação	4	Linguagem de Programação II	3	20	0
Ciência da Computação	4	Pesq. de Prática em Ciência da Computação II	1	20	0
Ciência da Computação	4	Sistemas Operacionais	4	20	0
Ciência da Computação	4	Tópicos Especiais I	2	20	0
Ciência da Computação	5	Banco de Dados II	2	20	0
Ciência da Computação	5	Compiladores	4	20	0
Ciência da Computação	5	Eletiva III	4	20	0
Ciência da Computação	5	Engenharia de Software II	4	15	0

Ciência da Computação	5	Estágio III	2	20	0
Ciência da Computação	5	Linguagem de Programação III	3	20	0
Ciência da Computação	5	Pesquisa de Prática em Ciência da Computação III	1	20	0
Ciência da Computação	5	Realidade Virtual	2	20	0
Ciência da Computação	6	Estágio IV	2	20	0
Ciência da Computação	6	Inteligência Artificial	4	20	0
Ciência da Computação	6	Pesquisa de Prática em Ciência da Computação IV	1	20	0
Ciência da Computação	6	Processamento de Imagens e Sinais	4	20	0
Ciência da Computação	6	Redes e Sistemas Distribuídos	3	20	0
Ciência da Computação	6	Segurança e Auditoria de Sistemas	4	20	0
Ciência da Computação	7	Computação Gráfica	4	20	0
Ciência da Computação	7	Estágio V	1	20	0
Ciência da Computação	7	Ética e Cultura Religiosa	4	20	0
Ciência da Computação	7	Pesquisa de Prática em Ciência da Computação V	1	20	0
Ciência da Computação	7	Teoria dos Grafos	4	20	0
Ciência da Computação	7	Tópicos Especiais II	1	20	0
Ciência da Computação	7	Trabalho de Conclusão de Curso I	2	20	0
Ciência da Computação	8	Estágio VI	2	20	0
Ciência da Computação	8	Linguagens Formais, Autômatos e Computabilidade	2	20	0
Ciência da Computação	8	Pesquisa de Prática em Ciência da Computação VI	1	20	0
Ciência da Computação	8	Redes Neurais	2	20	0
Ciência da Computação	8	Sociologia da Responsabilidade Social	4	20	0
Ciência da Computação	8	Tópicos Especiais III	3	20	0
Ciência da Computação	8	Trabalho de Conclusão de Curso II	2	20	0

Figura 35 - Relação Dados Curso - Disciplina.

Fonte: Elaborada pelo Autor (2013).

A Figura 36 demonstra a relação das disciplinas vinculadas a cada professor, bem como a quantidade de créditos.

Professor	Disciplina	Qtde. Créditos
Anderson Talon	Algoritmos	4
Anderson Talon	Estrutura de Dados	4
André Castro	Circuitos Digitais	2
André Castro	Eletiva I	4
André Castro	Estrutura de Dados	4
André Castro	Introdução à Lógica	4
André Castro	Linguagem de Programação I	3
André Castro	Sistemas Operacionais	4
André Castro	Técnicas de Programação	4
Elaine Gatto	Algoritmos	4
Elaine Gatto	Linguagem de Programação I	3

Elaine Gatto	Linguagem de Programação II	3
Elaine Gatto	Linguagem de Programação III	3
Elaine Gatto	Pesq. de Prática em Ciência da Computação I	1
Elaine Gatto	Pesq. de Prática em Ciência da Computação II	1
Elaine Gatto	Pesquisa de Prática em Ciência da Computação III	1
Elaine Gatto	Pesquisa de Prática em Ciência da Computação IV	1
Elaine Gatto	Pesquisa de Prática em Ciência da Computação V	1
Elvio Gilberto Silva	Arquitetura de Computadores	2
Elvio Gilberto Silva	Eletiva I	4
Elvio Gilberto Silva	Engenharia de Software I	2
Elvio Gilberto Silva	Engenharia de Software II	4
Elvio Gilberto Silva	Introdução à Computação	4
Elvio Gilberto Silva	Introdução à Lógica	4
Elvio Gilberto Silva	Métodos e Técnicas de Pesquisa	4
Elvio Gilberto Silva	Pesquisa de Prática em Ciência da Computação I	1
Elvio Gilberto Silva	Pesquisa de Prática em Ciência da Computação II	1
Elvio Gilberto Silva	Pesquisa de Prática em Ciência da Computação III	1
Elvio Gilberto Silva	Pesquisa de Prática em Ciência da Computação IV	1
Elvio Gilberto Silva	Técnicas de Programação	4
Elvio Gilberto Silva	Tópicos Especiais II	1
Elvio Gilberto Silva	Tópicos Especiais III	3
Elvio Gilberto Silva	Trabalho de Conclusão de Curso I	2
Elvio Gilberto Silva	Trabalho de Conclusão de Curso II	2
Erica Paveloski	Eletiva I	4
Erica Paveloski	Eletiva III	4
Erica Paveloski	Estágio II	1
Erica Paveloski	Fundamentos de Administração	4
Erica Paveloski	Fundamentos de Sistema de Informação	2
Erica Paveloski	Métodos e Técnicas de Pesquisa	4
Fátima Araújo	Cálculo: Integração de uma variável	4
Fátima Araújo	Cálculo: Limite e Derivada	4
Fátima Araújo	Estatística	2
Fátima Araújo	Introdução à Lógica	4
Henrique Pachioni Martins	Arquitetura de Computadores	2
Henrique Pachioni Martins	Banco de Dados I	4
Henrique Pachioni Martins	Circuitos Digitais	2
Henrique Pachioni Martins	Eletiva I	4
Henrique Pachioni Martins	Eletiva III	4
Henrique Pachioni Martins	Estágio III	2
Henrique Pachioni Martins	Fundamentos de Administração	4
Henrique Pachioni Martins	Fundamentos de Sistema de Informação	2
Henrique Pachioni Martins	Introdução à Computação	4
Henrique Pachioni Martins	Pesquisa de Prática em Ciência da Computação I	1
Henrique Pachioni Martins	Pesquisa de Prática em Ciência da Computação II	1
Henrique Pachioni Martins	Pesquisa de Prática em Ciência da Computação III	1

Henrique Pachioni Martins	Pesquisa de Prática em Ciência da Computação IV	1
Henrique Pachioni Martins	Redes e Sistemas Distribuídos	3
Henrique Pachioni Martins	Segurança e Auditoria de Sistemas	4
Henrique Pachioni Martins	Sistemas Operacionais	4
José Rafael	Ética e Cultura Religiosa	4
José Rafael	Sociologia da Responsabilidade Social	4
Kelton Augusto Pontara	Arquitetura de Computadores	2
Kelton Augusto Pontara	Circuitos Digitais	2
Kelton Augusto Pontara	Compiladores	4
Larissa Pavarini	Algoritmos	4
Larissa Pavarini	Realidade Virtual	2
Larissa Pavarini	Redes Neurais	2
Larissa Pavarini	Técnicas de Programação	4
Larissa Pavarini	Teoria dos Grafos	4
Márcia Marinho	Cálculo: Integração de uma variável	4
Márcia Marinho	Cálculo: Limite e Derivada	4
Márcia Marinho	Estatística	2
Márcio Cardim	Banco de Dados I	4
Márcio Cardim	Banco de Dados II	2
Márcio Cardim	Eletiva I	4
Márcio Cardim	Introdução à Computação	4
Márcio Cardim	Técnicas de Programação	4
Maria Helena	Comunicação e Expressão	4
Patrícia Bellin	Computação Gráfica	4
Patrícia Bellin	Eletiva I	4
Patrícia Bellin	Estágio III	2
Patrícia Bellin	Linguagem de Programação III	3
Patrícia Bellin	Linguagens Formais, Autômatos e Computabilidade.	2
Patrícia Bellin	Pesquisa de Prática em Ciência da Computação I	1
Patrícia Bellin	Pesquisa de Prática em Ciência da Computação II	1
Patrícia Bellin	Pesquisa de Prática em Ciência da Computação III	1
Patrícia Bellin	Pesquisa de Prática em Ciência da Computação IV	1
Patrícia Bellin	Processamento de Imagens e Sinais	4
Patrícia Bellin	Redes Neurais	2
Patrícia Bellin	Técnicas de Programação	4
Patrícia Bellin	Teoria dos Grafos	4
Patrick Pedreira Silva	Algoritmos	4
Patrick Pedreira Silva	Banco de Dados I	4
Patrick Pedreira Silva	Estrutura de Dados	4
Patrick Pedreira Silva	Inteligência Artificial	4
Patrick Pedreira Silva	Introdução à Computação	4
Patrick Pedreira Silva	Introdução à Lógica	4
Patrick Pedreira Silva	Linguagem de Programação I	3
Patrick Pedreira Silva	Realidade Virtual	2
Patrick Pedreira Silva	Redes Neurais	2

Patrick Pedreira Silva	Sistemas Operacionais	4
Patrick Pedreira Silva	Técnicas de Programação	4
Patrick Pedreira Silva	Teoria dos Grafos	4
Patrick Pedreira Silva	Trabalho de Conclusão de Curso I	2
Patrick Pedreira Silva	Trabalho de Conclusão de Curso II	2
Rosane Araújo	Cálculo: Integração de uma variável	4
Rosane Araújo	Cálculo: Limite e Derivada	4
Rosane Araújo	Estatística	2
Rosane Araújo	Introdução à Lógica	4
Rosane Araújo	Métodos e Técnicas de Pesquisa	4
William Galvão	Eletiva I	4
William Galvão	Tópicos Especiais I	2
William Galvão	Tópicos Especiais II	1
William Galvão	Tópicos Especiais III	3

Figura 36 - Relação Professor-Disciplina.

Fonte: Elaborada pelo Autor (2013).

A plataforma de *hardware* utilizada para a realização dos testes do protótipo consistiu de um microcomputador com as seguintes características:

- Notebook Intel® Core i5-2450M com processador de 2.50GHz e 4 Gb de RAM.

Para fins de testes do Algoritmo, as tabelas no Banco de Dados foram populadas de acordo com a grade curricular vigente para os alunos do 4º ano de Ciência da Computação, conforme mencionado na Figura 35. As etapas a seguir demonstram os testes realizados:

A Figura 37 ilustra os parâmetros utilizados para a geração do horário, o qual faz referência ao 1º semestre do curso de Ciência da Computação englobando todas as disciplinas.

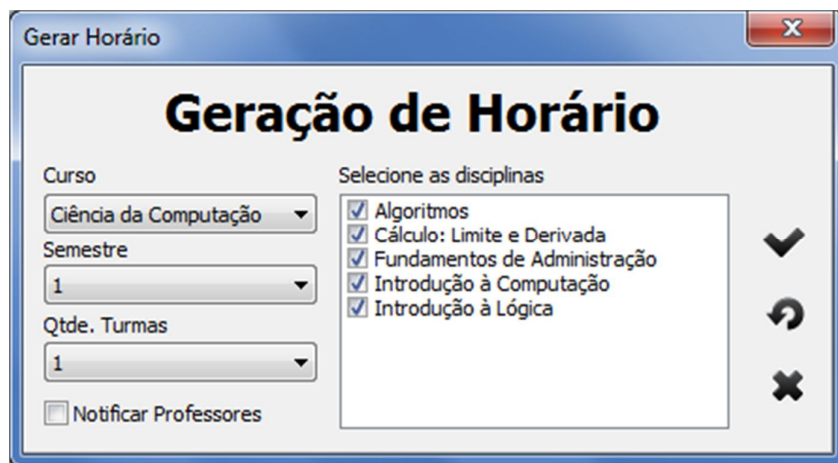


Figura 37 - Geração de Horário. 1ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 38 é possível identificar todas as informações que estão vinculadas ao horário a ser gerado.

Professor	Disciplina	S	T	Q	Q	S	S	F	Rest. (%)	Qtde. Créd.
Erica Paveloski	Fundamentos de Administração	1	0	0	0	1	0	1	26,5	4
Márcia Marinho	Cálculo: Limite e Derivada	1	0	1	0	1	0	1	35	4
Anderson Talon	Algoritmos	1	0	1	1	1	0	0	43	4
André Castro	Introdução à Lógica	1	0	1	1	1	0	0	43	4
Márcio Cardim	Introdução à Computação	1	1	0	1	1	1	0	51,5	4
Elvio Gilberto Silva	Introdução à Computação	1	1	0	1	1	0	0	53	4
Elvio Gilberto Silva	Introdução à Lógica	1	1	0	1	1	0	0	53	4
Fátima Araújo	Cálculo: Limite e Derivada	1	1	0	1	1	0	1	53	4
Fátima Araújo	Introdução à Lógica	1	1	0	1	1	0	0	53	4
Henrique Pachioni Martins	Fundamentos de Administração	1	1	0	0	1	1	1	53	4
Henrique Pachioni Martins	Introdução à Computação	1	1	0	0	1	1	0	53	4
Rosane Araújo	Cálculo: Limite e Derivada	1	1	1	0	1	0	1	53	4
Rosane Araújo	Introdução à Lógica	1	1	1	0	1	0	0	53	4
Patrick Pedreira Silva	Algoritmos	1	1	1	1	0	0	0	63	4
Patrick Pedreira Silva	Introdução à Computação	1	1	1	1	0	0	0	63	4
Patrick Pedreira Silva	Introdução à Lógica	1	1	1	1	0	0	0	63	4

Figura 38 - Geração de Horário. 2ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Antes de inicializar o algoritmo, é realizada uma validação que consiste em uma análise prévia para verificar se realmente é possível gerar o horário. A validação considera os seguintes fatores:

- A quantidade de professores vinculados é maior que a quantidade de disciplinas?
- Dentre as disciplinas selecionadas, há pelo menos um professor vinculado?

Se a validação retornar 'True' significa que é possível gerar um horário, caso contrário, é exibida uma mensagem informando os motivos da impossibilidade da geração do horário. Após a validação, é dada continuidade no processo. A Figura 39 representa a terceira etapa deste processo, que consiste na alocação dos dados resultado da cláusula SELECT para os objetos, que armazenam as informações de cada professor.

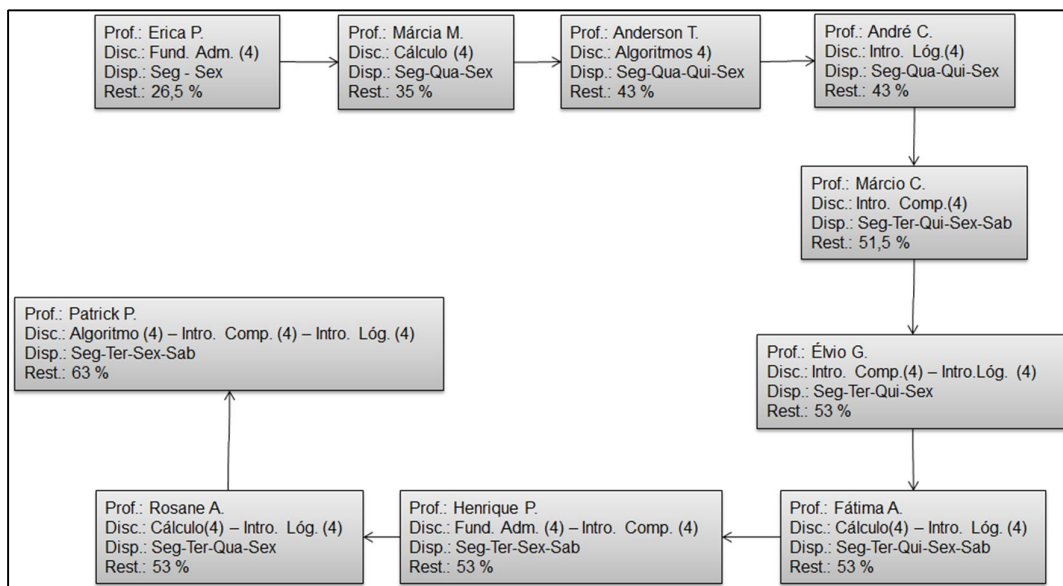


Figura 39 - Geração de Horário. 3ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 40, é iniciado o processo de geração do horário, respeitando ao percentual de restrição de cada professor, desta maneira a professora Erica com 26,5% de restrição preenche a segunda-feira com a disciplina de 'Fundamentos de Administração'. A partir deste preenchimento é realizada a verificação prévia nos demais objetos, onde são atualizados as disciplinas e os dias disponíveis de cada professor.

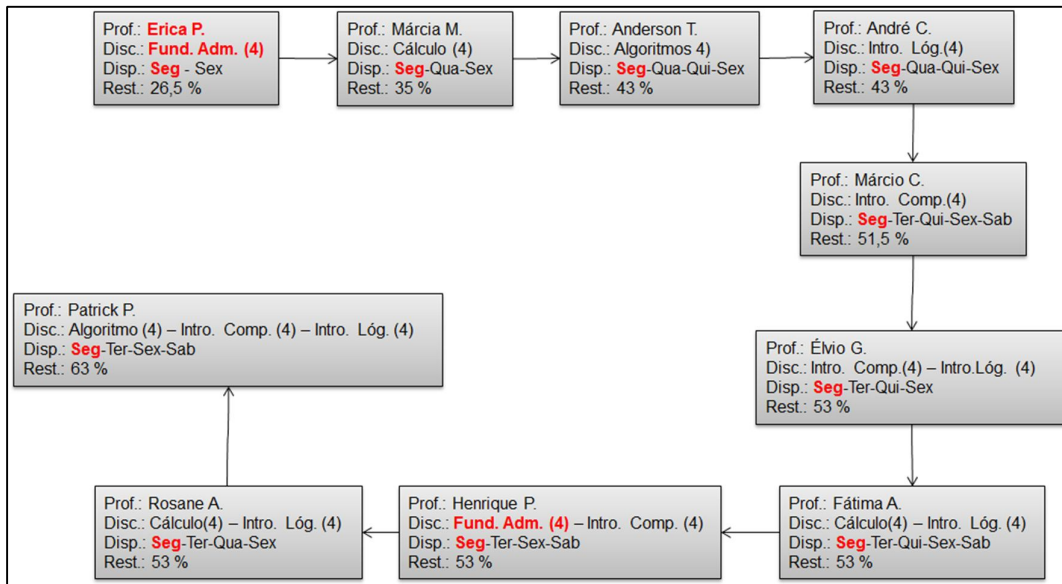


Figura 40 - Geração de Horário. 4ª Etapa.

Fonte: Elaborada pelo Autor (2013).

A Figura 41 representa a sequência do processo, em que a professora Márcia, com 35% de restrição, preenche a quarta-feira com a disciplina de 'Cálculo: Limite e Derivada'.

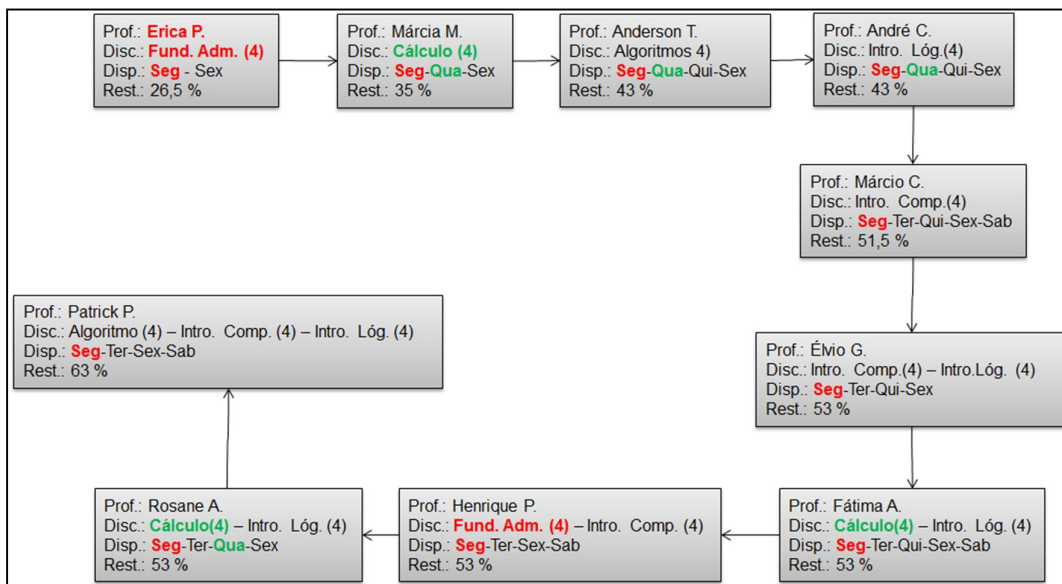


Figura 41 - Geração de Horário. 5ª Etapa.

Fonte: Elaborada pelo Autor (2013).

Na Figura 42, há dois professores com o mesmo percentual de restrição (43%), nesta situação é respeitada a ordem alfabética com relação ao nome dos professores, sendo assim, o professor Anderson preenche a quinta-feira com a disciplina de ‘Algoritmos’.

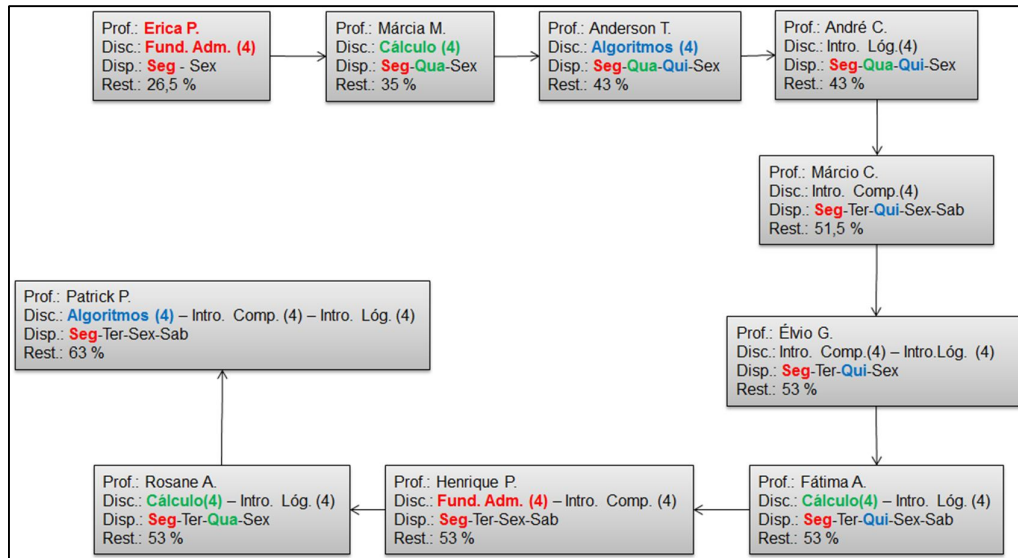


Figura 42 - Geração de Horário. 6ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 43, o professor André com os mesmos 43% de restrição preenche a sexta-feira com a disciplina de ‘Introdução à Lógica’.

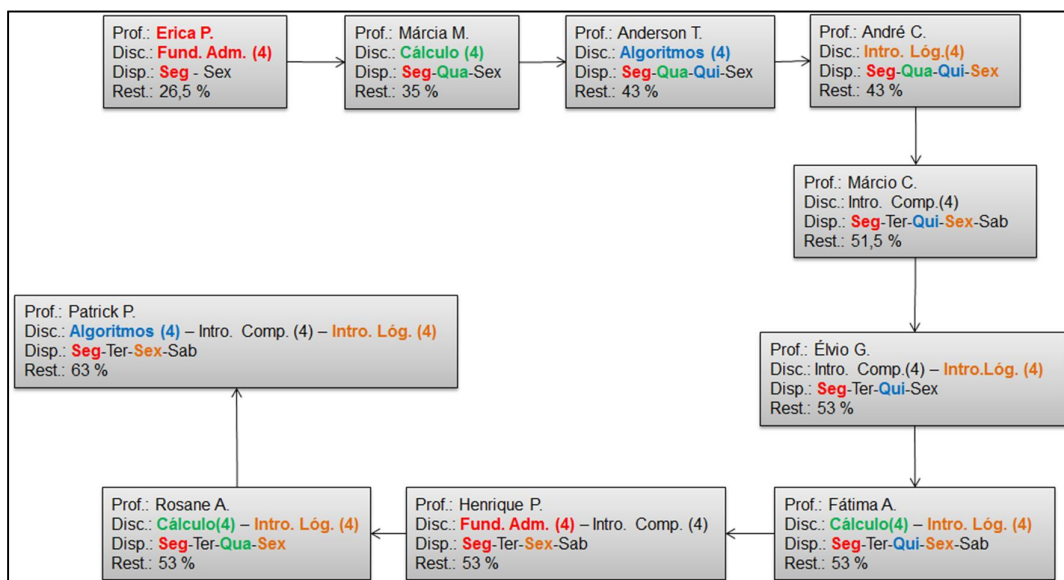


Figura 43 - Geração de Horário. 7ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 44, por fim o professor Márcio, com 51,5% de restrição, preenche a terça-feira com a disciplina de 'Introdução a Computação'.

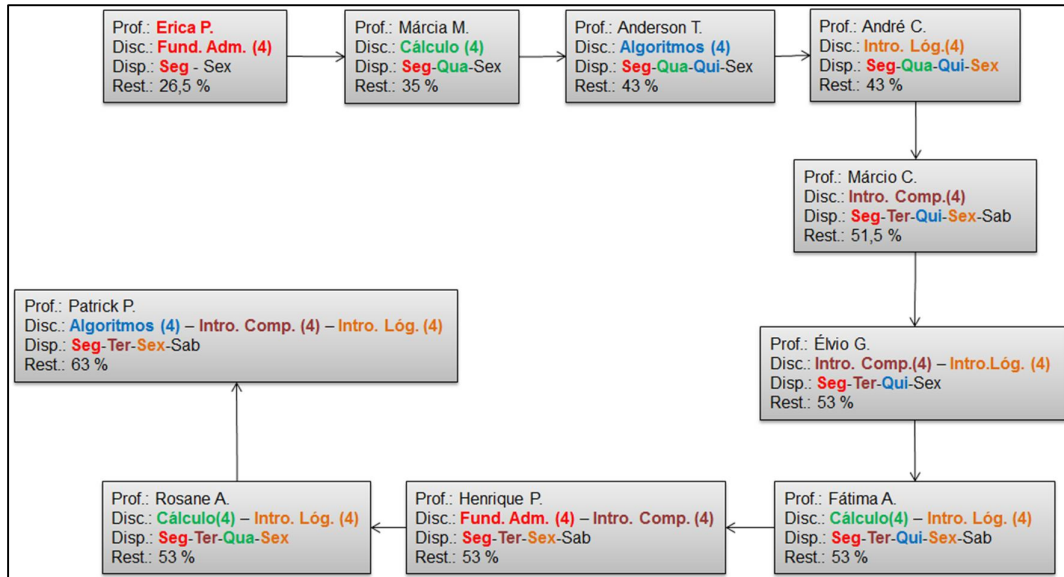


Figura 44 - Geração de Horário. 8ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 45, é possível visualizar o horário final com as disciplinas selecionadas no início no processo. As restrições impostas foram sanadas, uma vez que, a heurística tratou as informações de forma eficiente evitando um trabalho custoso com relação ao número de tentativas.

Horário	Segunda	Terça	Quarta	Quinta	Sexta
19:00 - 19:45	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.
19:45 - 20:30	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.
20:45 - 21:30	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.
21:30 - 22:15	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.

Figura 45 - Horário Final. 9ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Neste horário gerado, é possível visualizar todo o processo:

- Preenchimento dos parâmetros na *interface* gráfica;
- Geração do SELECT com a heurística adotada;
- Validação;
- Se validação for verdadeira inicializa o Algoritmo senão exibe mensagem;

Se porventura no preenchimento dos parâmetros a quantidade de turmas for igual a 2, é analisada a questão das disciplinas 'fixas', conforme mencionado na seção 4. Nesta situação, as disciplinas 'Fundamentos de Administração' e 'Cálculo: Limite e Derivada' exercem um papel de espelho do 1º para o 2º horário, resultando na Figura 46:

Horário	Segunda	Terça	Quarta	Quinta	Sexta
19:00 - 19:45	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.
19:45 - 20:30	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.
20:45 - 21:30	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.
21:30 - 22:15	Fund. Adm.	Intro. Comp.	Márcia M.	Algoritmos	Intro. Lóg.
	Erica P.	Márcio C.	Cálculo	Anderson T.	André C.

Horário	Segunda	Terça	Quarta	Quinta	Sexta	Sábado
19:00 - 19:45	Fund. Adm.		Márcia M.			
	Erica P.		Cálculo			
19:45 - 20:30	Fund. Adm.		Márcia M.			
	Erica P.		Cálculo			
20:45 - 21:30	Fund. Adm.		Márcia M.			
	Erica P.		Cálculo			
21:30 - 22:15	Fund. Adm.		Márcia M.			
	Erica P.		Cálculo			

Figura 46 - Análise disciplinas 'fixas'.

Fonte: Elaborada pelo Autor (2013).

Nota-se que antes de inicializar o 2º horário, as disciplinas consideradas 'fixas' são automaticamente preenchidas, logo depois de realizado a verificação prévia, os dias disponíveis são: terça, quinta, sexta e sábado. Nota-se na Figura 47, que os professores Anderson, André e Márcio não possuem

disponibilidade na quinta, sexta e terça respectivamente, pois possuem o vínculo com o 1º horário. Prosseguindo, temos os seguintes objetos:

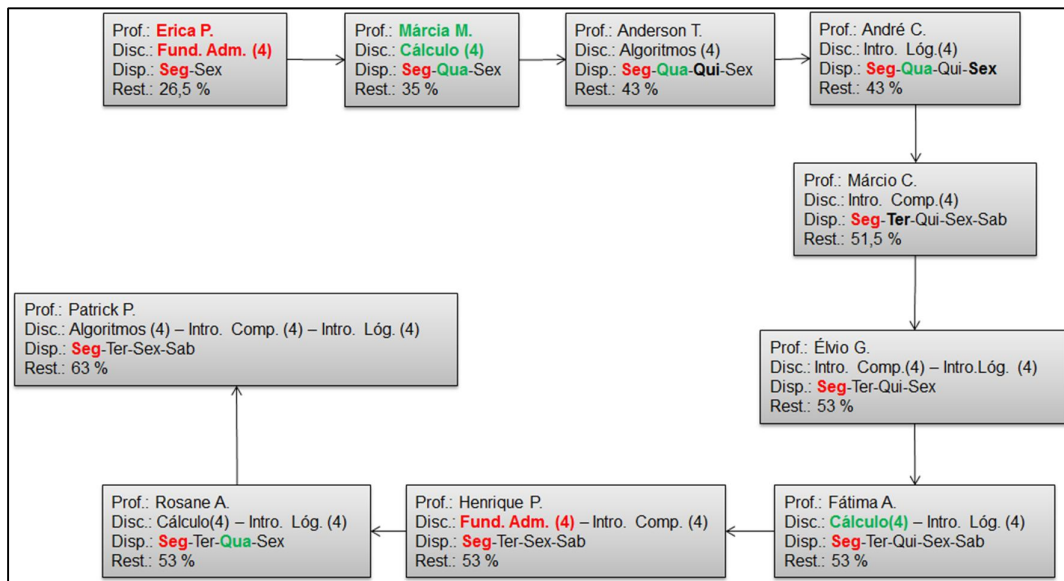


Figura 47 - Geração Horário. 9ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 48, o professor Anderson com 43% de restrição preenche a sexta-feira com a disciplina de 'Algoritmos'.

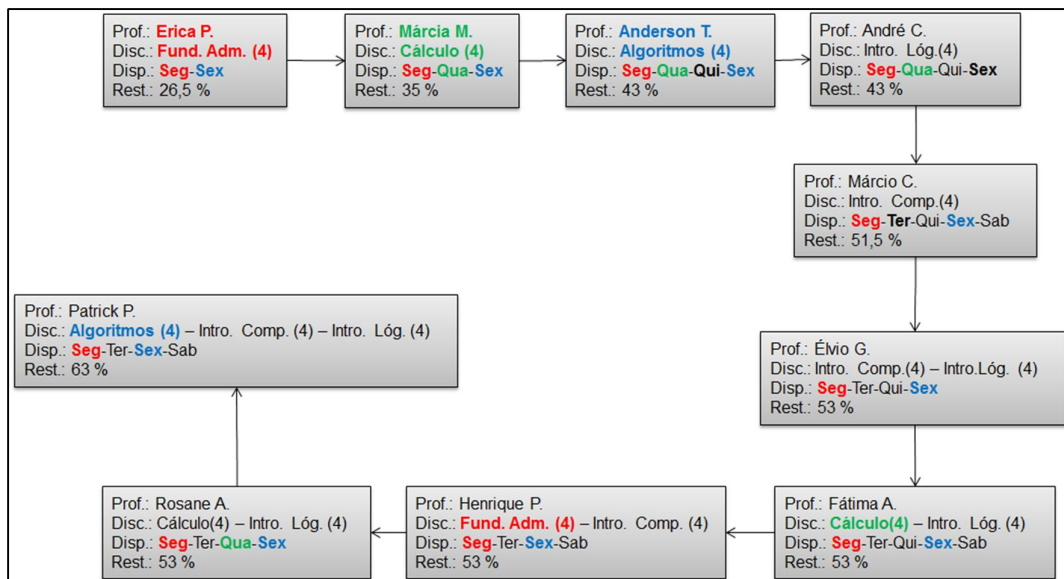


Figura 48 - Geração Horário. 10ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 49, o professor André com 43% preenche a quinta-feira com a disciplina de 'Introdução a Lógica'.

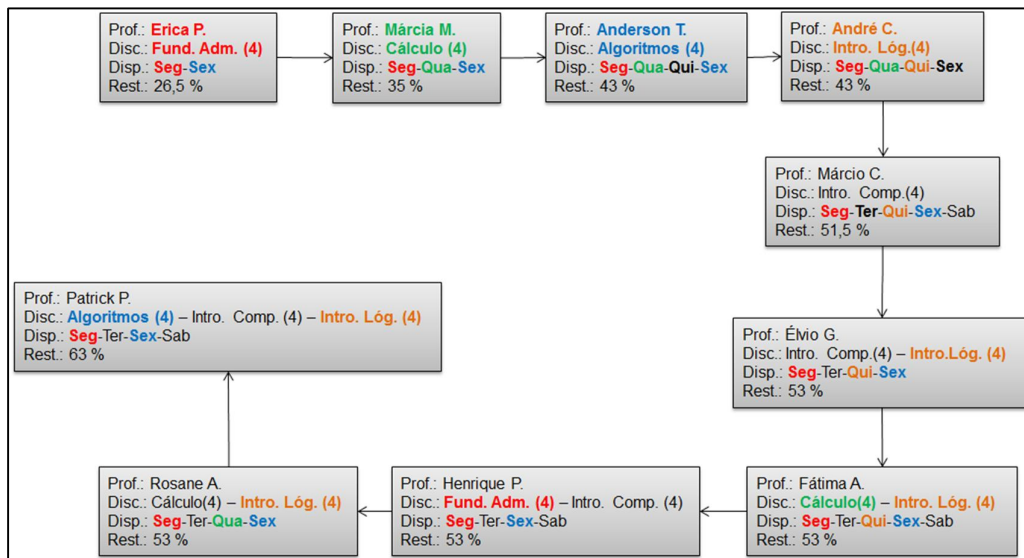


Figura 49 - Geração Horário. 11ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 50, o professor Elvio com 53% preenche a terça-feira com a disciplina de 'Introdução a Computação'. Nota-se nesta etapa que o próximo professor com nível de restrição maior é o professor Márcio, porém seu único dia disponível é o sábado, nesta situação o algoritmo procura o próximo professor com disponibilidade entre segunda à sexta, para que assim seja respeitada a restrição imposta na seção 4, que prioriza o horário de segunda à sexta.

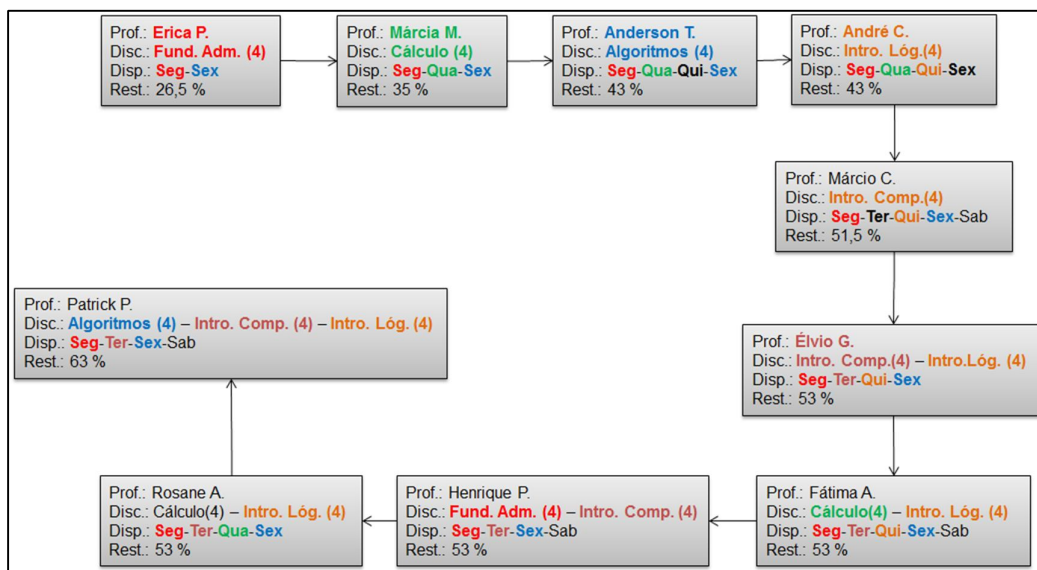


Figura 50 - Geração Horário. 12ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Na Figura 51, tem-se o seguinte horário:

Horário	Segunda	Terça	Quarta	Quinta	Sexta
19:00 - 19:45	Fund. Adm.	Intro. Comp.	Márcia M.	Intro. Lóg.	Algoritmos
	Erica P.	Elvio G.	Cálculo	André C.	Anderson T.
19:45 - 20:30	Fund. Adm.	Intro. Comp.	Márcia M.	Intro. Lóg.	Algoritmos
	Erica P.	Elvio G.	Cálculo	André C.	Anderson T.
20:45 - 21:30	Fund. Adm.	Intro. Comp.	Márcia M.	Intro. Lóg.	Algoritmos
	Erica P.	Elvio G.	Cálculo	André C.	Anderson T.
21:30 - 22:15	Fund. Adm.	Intro. Comp.	Márcia M.	Intro. Lóg.	Algoritmos
	Erica P.	Elvio G.	Cálculo	André C.	Anderson T.

Figura 51 - Horário Final. 13ª Etapa.
Fonte: Elaborada pelo Autor (2013).

Fica evidente que o todo processo realizado de forma automática é mais vantajoso do que o processo manual, pois a informação é tratada mais rápida.

7 CONSIDERAÇÕES FINAIS

O propósito deste trabalho foi demonstrar que o Algoritmo de Satisfação de Restrições (PSR) é eficaz no ato da geração de horários escolares. O trabalho manual requer tempo, e no que diz respeito à geração de horários envolve diversas possibilidades, resultando em um processo muito custoso e trabalhoso, deste modo é possível concluir que o Algoritmo processa as informações de forma mais rápida e eficiente, colaborando com a redução de tempo.

Ao decorrer deste trabalho foram encontradas diversas dificuldades, pois para o desenvolvimento não foi utilizado nenhum Framework, desta maneira as informações foram tratadas mediante ao Algoritmo proposto.

Para trabalhos futuros, a aplicação possui alguns aspectos que podem ser aperfeiçoados, como:

- Quantidade de turmas ilimitada no ato da geração do horário;
- Alterar a aplicação de desktop para Web, para que o usuário tenha maior praticidade para manipular as informações;
- No cadastro de disponibilidade do professor, mencionar os horários disponíveis com mais detalhes.

Em síntese o trabalho atingiu o resultado esperado, demonstrando que para gerar um horário escolar é necessário respeitar a uma série de restrições cujo Algoritmo de Satisfação de Restrições possui todo um conceito que auxilia na manipulação destas inter-relações professores, disponibilidade e disciplinas.

REFERÊNCIAS

ANDERSON, J.A. and Rosenfeld, E. (Eds). (1988). Neurocomputing: Foundations of Research. MIT Press, Cambridge, Massachusetts.

CORDENONSI, A. Z. ; ARAMBURU, L. G. C. ; ALMANCA, L. Resolução do Problema de Quadro de Horários Através de um Algoritmo de Satisfação de Restrições. In: VIII Simpósio de Informática e III Mostra de Software Acadêmico, 2003, Uruguaina- RS, 2003.

FERNANDES, Anita Maria da Rocha. Inteligência Artificial, Noções Gerais (2003).

GUARDA, Álvaro. CIC250 - Inteligência Artificial. Ouro Preto - Mg: Universidade Federal de Ouro Preto, 2006. 42 p. Disponível em: <<http://pt.scribd.com/doc/62492025/20/Subida-da-encosta-Hill-Climbing>>. Acesso em: 19 maio 2013

LOBO, Eduardo Luiz Miranda. UMA SOLUÇÃO DO PROBLEMA DE HORÁRIO ESCOLAR VIA ALGORITMO GENÉTICO PARALELO. 2005. 95 f. Dissertação (Mestre) - Curso de Mestrado em Modelagem Matemática e Computacional, Departamento de Pesquisa e Pós-graduação Curso de Mestrado em Modelagem Matemática e Computacional, Centro Federal De Educação Tecnológica De Minas Gerais, Belo Horizonte, 2005.

QUINET, Marcos. Inteligência Artificial. Rio de Janeiro: Marcos Quinet, 2013. 64 p. Disponível em: <http://www.professores.uff.br/mquinet/06_IA.pdf>. Acesso em: 28 maio 2013.

MORAES, Fernando. Cérebro virtual simula comportamento e realiza tarefas cognitivas. Folha de São Paulo: Ciência, São Paulo, 04 dez. 2012. p. 1-1. Disponível em: <<http://www1.folha.uol.com.br/ciencia/1195388-cerebro-virtual-simula-comportamento-e-realiza-tarefas-cognitivas.shtml>>. Acesso em: 27 maio 2013.

RUSSELL, Stuart e NORVIG, Peter. 2004. Inteligência Artificial. 2º Edição. Rio de Janeiro : Elsevier, 2004. p. 992.

SANTANA, Ana Lucia. **Behaviorismo**. Disponível em: <<http://www.infoescola.com/psicologia/behaviorismo/>>. Acesso em: 19 maio 2013.

WINSTON, Patrick Henry. 1987. Inteligência Artificial. Rio de Janeiro: Livros Técnicos e Científicos, 1987.