



Universidade Eduardo Mondlane



Faculdade de Ciências

Departamento de Matemática e Informática

Curso: Licenciatura em Informática

Tema: Programação de Horários Escolares

Estudo de caso:

Escola Secundaria da Zona Verde

Escola Primaria Completa 1 de Junho

Autor: Carlos Augusto Caimane



Universidade Eduardo Mondlane



Faculdade de Ciências

Departamento de Matemática e Informática

Trabalho de Licenciatura

Tema: Programação de Horários Escolares

Estudo de caso:

Escola Secundária da Zona Verde

Escola Primária Completa 1 de Junho

Supervisor: Prof. Doutor Emílio Mosse

Autor: Carlos Augusto Caimane

Maputo, Novembro de 2014

Dedicatória

Dedico à minha mãe e aos meus irmãos, Herculano Augusto Inguane e a Rita Augusto pelo apoio, encorajamento e paciência que me concederam ao longo da minha carreira.

Agradecimentos

- 🚩 Agradeço primeiramente ao meu supervisor Prof. Doutor Emílio Mosse pela supervisão intelectual, paciência, atenção, críticas e ensinamentos por ele dados ao longo do trabalho;
- 🚩 A Deus, a quem devemos por tudo e por todos.
- 🚩 A todos os que directa ou indirectamente me ajudaram a concretizar este trabalho.
- 🚩 Um agradecimento especial vai ao Sr. Adamo Fakira Nunu que me prestou apoio sempre que eu precisei;

Declaração de honra

Declaro por minha honra, que este trabalho é fruto da minha investigação, e que o mesmo foi realizado para ser submetido à Faculdade de Ciências, em cumprimento parcial, para obtenção do grau de Licenciatura em Informática na Universidade Eduardo Mondlane.

Maputo, Novembro de 2014

(Carlos Augusto Caimane)

Resumo

A planificação de um ano lectivo de qualquer instituição de ensino compreende três fases funcionais, nomeadamente: a preparatória, a implementação e a avaliação do seu desempenho.

Na fase preparatória, uma das actividades incontornáveis e imprescindíveis é a elaboração de horários, instrumento através do qual se regula o funcionamento e a gestão do tempo lectivo nas escolas. É sobre esta fase que o presente trabalho se ocupa.

Durante o período preparatório, as direcções pedagógicas das escolas ocupam-se, dentre outras actividades, com a elaboração de horários lectivos. Trata-se de uma actividade complexa que exige muito zelo aquando da sua execução, dado que tem por função a distribuição dos tempos lectivos pelos docentes, respeitando o princípio de impenetrabilidade (aqui designado sobreposição de disciplinas).

No entanto, esta monografia propõe-se a apresentar um modelo de programa informático de geração de horários, utilizando a combinação das meta-heurísticas, busca tabu e busca local iterada, que possa facilitar e dinamizar este exercício.

Palavras-chave: *Planificação; fase preparatória; elaboração; horário; gestão de tempo; software.*

Dedicatória.....	I
Agradecimentos.....	II
Declaração de honra.....	III
Resumo.....	IV
Índice de Tabelas.....	2
Glossário.....	4
1. Introdução.....	5
1.1 Definição do problema.....	6
1.2 Objectivos.....	7
2. Metodologia do trabalho.....	8
2.1 Metodologia de Pesquisa.....	9
2.2 Metodologia de desenvolvimento.....	11
3. Fundamentação teórica.....	14
Métodos Heurísticos.....	17
3.1 Busca Local.....	17
3.2 Busca Tabu (BT).....	20
3.3 Algoritmos Evolutivos.....	24
3.4 Métodos Baseados em Programação Linear.....	26
3.5 Recozimento Simulado (Simulated Annealing).....	27
4. Estudo de Caso.....	28

4.1	Situação actual.....	30
5.	Análise e discussão de dados.....	31
5.1	Modelo Proposto.....	36
6.	Conclusões e recomendações.....	44
	Referências Bibliográficas.....	46
7.	Anexos.....	49
7.1	Guião de entrevista.....	49
7.2	Interface gráfica.....	50

Índice de figuras

Figura 1 - Pontos da função de busca local (Fonte: Stuart Russel e Peter Norving (2007)).....	18
Figura 2 - Solução inicial (Adaptado de: Stuart Russel e Peter Norving (2007)).....	20
Figura 3 - Ótimo Local (Adaptado de: Stuart Russel e Peter Norving (2007)).....	21
Figura 4 - Elementos Tabu (Adaptado de: Stuart Russel e Peter Norving (2007)).....	21
Figura 5 - Ótimo Global. (Adaptado de: Stuart Russel e Peter Norving (2007)).....	22
Figura 6 - Pseudocódigo do algoritmo evolutivo (Fonte: Benz (2005)).....	25
Figura 7 – Duração da elaboração de horários.....	35
Figura 8 – Grau de dificuldade de elaboração de horários.....	35
Figura 9 – Satisfação de professores pelos horários.....	36
Figura 10 - Diagrama de classes.....	37
Figura 11 – Caso de uso do administrador e do professor.....	38
Figura 12 – Caso de uso de gestão de utilizadores.....	39
Figura 13 – Casos de uso de gestão de cursos e de disciplinas.....	39
Figura 14 - Casos de uso.....	40
Figura 15 - Diagrama de sequência de eventos para registrar a disciplina.....	41
Figura 16 - Diagrama de sequência de eventos para registrar o curso.....	41
Figura 17 - Diagrama de sequência de eventos para registrar o professor.....	42
Figura 18 - Diagrama de sequência de eventos para registrar a sala.....	42
Figura 19 - Diagrama de sequência de eventos para o registo da turma.....	43
Figura 20 - Diagrama de sequência de eventos para o registo do horário.....	43
Figura 21 – Autenticação de utilizadores.....	50
Figura 22 – Tela principal.....	51
Figura 23 – Formulário de registo de professores.....	52
Figura 24 – Formulário de registo de disciplinas.....	53
Figura 25 – Formulário de registo de turmas.....	54
Figura 26 - Formulário de registo de salas.....	55
Figura 27 – Formulário de registo de cursos.....	55
Figura 28 – Criação de horários.....	56
Figura 29 – Exemplo de horário de um dos professores.....	57
Figura 30 – Exemplo de horário de uma das turmas.....	57

Índice de Tabelas

Tabela 1 - Símbolos e acrónimos.....	4
Tabela 2 - Relação das turmas usadas na simulação de elaboração de horários.....	10
Tabela 3 - Algoritmo de busca local. (Adaptado de Costa F.P. (2003)).....	19
Tabela 4 - Procedimento Busca Tabu (Fonte: Costa F.P.(2003)).....	24
Tabela 5 - Distribuição das classes de EP1 por turno da Escola Primária Completa 1 de Junho...	28
Tabela 6 - Distribuição das classes de EP2 por turno da Escola Primária Completa 1 de Junho...	29
Tabela 7 - Distribuição das classes por turno da ESZV.....	29
Tabela 8 - Distribuição de horários nos diferentes turnos.....	32
Tabela 9 - Distribuição das classes por turno.....	33
Tabela 10 – Relação de cargas horárias do ensino secundário.....	33
Tabela 11 – Relação de cargas horárias do ensino primário.....	34

Glossário

Tabela 1 - Símbolos e acrónimos

Notação	Descrição
ESZV	Escola Secundária da Zona Verde
EPC	Escola Primária Completa
BD	Base de dados
BT	Busca Tabu
EP1	Ensino Primário do Primeiro Grau
EP2	Ensino Primário do Segundo Grau
MySQL	<i>My structured Query Language</i> (Linguagem estruturada de consultas).
SQL	É uma linguagem de definição, criação e manipulação de dados. Esta linguagem disponibiliza comandos usados tanto na definição e alteração de estruturas de dados assim como na manipulação de dados numa BD.
TIC	Tecnologias de Informação e Comunicação

EM	Universidade Eduardo Mondalane
----	--------------------------------

1. Introdução

Os sistemas baseados em Tecnologias de Informação e Comunicações (TIC), com mais destaque para os que se baseiam em computadores, tornam-se um veículo imprescindível na disponibilização de aplicações que facilitam a gestão das organizações, pela rapidez que oferecem no processamento de dados, por disponibilizarem informação de qualidade e em boa quantidade, para satisfazer as necessidades de negócio das organizações (Cung, 2002).

Segundo o Ministério da Educação e Cultura (2006), as TIC desempenham um importante papel de apoio ao sector da Educação, particularmente no reforço aos processos de gestão. Assim, há, actualmente, uma tendência de migração de sistemas manuais para os computadorizados, no sentido de tornar o trabalho das organizações mais rápido, o que implicará uma tomada rápida de decisões por parte destas. Um exemplo de adesão institucional no qual é possível consultar os resultados de exame de admissão sem necessidade de se deslocar até à faculdade pelos candidatos ao ingresso é o sistema *online* de gestão de pautas.

Apesar de haver organizações que já implementaram sistemas informáticos de gestão das suas actividades, há muitas ainda que carecem destes. Um desses produtos informáticos é o sistema de gestão de horários de trabalho.

De uma forma específica, o presente trabalho discute somente o problema de programação de horários escolares, que diz respeito à alocação das aulas de uma escola a um conjunto restrito de horários, sem descartar as restrições respectivas e assenta-se na automação do problema de programação de horários escolares, pois apesar de todo o avanço computacional, a maioria das escolas e algumas faculdades ainda realizam esta actividade de forma manual, tornando esta tarefa bastante demorada e ineficiente. Portanto, é objectivo principal deste trabalho, desenvolver uma aplicação que atenda as necessidades das escolas e de faculdades no que tange à gestão de horários, tentando desta forma incentivar o uso das TIC no sector da educação, para resolver alguns dos seus problemas.

1.1 Definição do problema

Mediante sondagens feitas nas Escolas Primária Completa 1 de Junho e Secundária da Zona Verde constatou-se que a elaboração de horários nestas é feita manualmente, exigindo o seguinte: Uma sala de aulas, três (3) cartolinas, giz, apagador e no mínimo dois (2) professores. Toda esta exigência, aliada à necessidade de se dever ter em conta algumas situações extra-docência, questões operacionais e pedagógicas relevantes e merecedoras de especial atenção, que afectam alguns professores, tornam lento, ineficiente e ineficaz a elaboração manual deste instrumento regulador do tempo lectivo.

O problema de elaboração de horários em escolas e em faculdades abordado neste trabalho consiste em um número p de professores para t turmas, d disciplinas para os p professores e h horários semanais nos quais decorrerão as aulas. A elaboração de uma grade horária escolar deve satisfazer as preferências dos professores, além das questões operacionais e pedagógicas. Assim, o trabalho torna-se muito complexo quando esta distribuição deve ser realizada manualmente e com pouco tempo para sua execução. As restrições para este tipo de problema, de uma forma geral, são as seguintes:

- **Sobreposição**

Em cada horário o professor não poderá ministrar aulas para mais de uma turma.

Em cada horário não poderá ter mais que um professor na turma;

Nenhuma disciplina poderá ter mais de um professor na mesma turma;

- **Carga horária**

A carga horária diária da turma deve ser satisfeita, bem como a carga horária semanal;

A carga horária semanal do professor estabelecida para cada uma das turmas deve ser contemplada;

A carga horária semanal de uma disciplina em cada turma deve ser satisfeita;

- **Disponibilidade**

Se o professor não estiver disponível para ensinar em um dado período, então não pode ser alocado a esse período.

- **Número de aulas**

Em uma disciplina de carga horária semanal de 3 horas a configuração deve ser de uma aula dupla e outra ministrada em um único tempo, ou o contrário;

Em disciplinas com mais de 3 horas as aulas devem ser dadas em duplas.

Aulas quebradas (aulas não consecutivas) de uma disciplina para uma turma não podem ser ministradas em um mesmo dia;

Diante do acima descrito, coube levantar, como pergunta-guia deste trabalho, a seguinte questão:

De que forma as escolas podem conceber os horários lectivos de modo a satisfazerem, em tempo útil, as necessidades básicas do seu funcionamento e gestão de tempo ao longo de um ano lectivo?

1.1.1 Hipóteses

Sendo a construção de horários lectivos nas escolas, uma tarefa difícil de se executar manualmente e um problema difícil de se solucionar por ser complexo e repleto de restrições, e mais difícil ainda é encontrar a melhor solução, que corresponda às preferências de todos os intervenientes, pode se minimizar estes constrangimentos e/ou alcançar a solução:

- Recorrendo a programas informáticos ou software concebidos para elaboração e distribuição de cargas horárias por professor
- Usando as TIC para desenvolver uma aplicação em java, para solucionar o problema em tempo certo.

1.2 Objectivos

1.2.1 Objectivo geral

Desenvolver um sistema de elaboração de horários escolares com recurso a linguagem de programação orientada a objectos java.

1.2.2 Objectivos específicos

Para atingir o objectivo geral anteriormente definido, definiram-se os seguintes objectivos específicos:

- Identificar os procedimentos e as restrições a respeitar na programação de horários escolares;
- Apresentar uma heurística que possa responder as exigências pedagógicas, assim como as preferências de cada professor com relação aos dias e horários de aulas;
- Construir uma heurística para o problema de programação de horários;

- Conceber o modelo de Sistema de elaboração de horários de escolas primárias e secundárias;
- Testar e implementar o Sistema de elaboração de horários;

2. Metodologia do trabalho

O uso de uma metodologia para a realização de uma actividade é um factor extremamente importante, pois permite modelar a actividade, bem como descrever correctamente os processos que irão se desencadear e, conseqüentemente, dá uma previsão aproximada do que será o resultado da actividade (Mangahela, 2012). Sendo assim, para se avaliar as dificuldades que as escolas têm no seu sistema actual de elaboração de horários (manual) e seguidamente construir uma heurística para solucionar o problema de programação de horários, para posteriormente desenvolver-se uma aplicação para executar esta tarefa, recorrer-se-á à seguinte metodologia de investigação:

2.1 Metodologia de Pesquisa

a. Tipo de estudo

Neste trabalho foram usadas pesquisas exploratórias, pois através destas foi possível proporcionar maior familiaridade com o problema de programação de horários através de entrevistas, pesquisas bibliográficas e observação, tomando em conta um estudo do caso.

Segundo Clemente et al (2007), uma pesquisa é considerada de natureza exploratória quando esta envolver levantamento bibliográfico, entrevistas com pessoas que tiveram ou que têm experiências práticas com o problema pesquisado e análise de exemplos que estimulem a compreensão. As pesquisas exploratórias visam proporcionar uma visão geral de um determinado facto, do tipo aproximativo.

Com recurso a este tipo de estudo, foi possível identificar os procedimentos e as restrições a respeitar na programação de horários de modo a permitir a construção de uma heurística para resolver este problema e o posterior desenvolvimento do sistema computacional.

b. Métodos de colheita de dados

Para permitir a recolha detalhada de toda a informação referente à programação de horários e através desta identificar os principais problemas enfrentados neste processo, foi necessário usar os seguintes métodos de colheita de dados:

Entrevistas semi-estruturadas

Segundo Bogdan & Biklen (2000), entrevistas semi-estruturadas permitem a recolha de dados comparáveis entre os vários sujeitos.

Nas entrevistas semi-estruturadas, os entrevistados podem trazer mais detalhes, em forma de conversação, às questões que lhes são colocadas, podendo ser introduzidas novas questões, caso o entrevistado suscite dúvidas no entrevistador (Lakatos, 2006).

Assim para permitir a melhor recolha de dados e identificar os requisitos necessários para o desenvolvimento do sistema, foram feitas entrevistas semi-estruturada a um universo de 20 intervenientes no processo de elaboração de horários, entre os quais professores e directores adjuntos pedagógicos:

- Aos professores envolvidos na elaboração de horários, em virtude de serem as pessoas que lidam directamente com a elaboração de horários.
- Aos directores adjuntos pedagógicos das escolas já referidas, pois eles participam na elaboração de horários, avaliam-nos e determinam se são exequíveis ou não para posteriormente divulgá-los. A avaliação é feita com base na verificação das restrições existentes na elaboração de horários.

Observação não participativa

Para além da informação dada pelos entrevistados, foi feita uma observação não participativa do processo de elaboração de horários e da ocupação das salas nas duas escolas. Neste processo foram observados 12 intervenientes (professores e o director adjunto pedagógico) das Escolas Primária Completa 1 de Junho e Secundária da Zona Verde a criarem tabelas nas quais são colocados os horários dos diferentes professores. Posteriormente, os mesmos foram observados a simularem uma elaboração de horários.

A simulação foi feita tomando um universo de 9 turmas com 3 secções distintas. A tabela 2 ilustra a relação de turmas e de secções usadas na simulação.

Tabela 2 - Relação das turmas usadas na simulação de elaboração de horários.

Quanto à ocupação das salas, foi observada a Directora Adjunta Pedagógica da Escola Secundária da Zona verde a simular a alocação das turmas na sala de informática.

A observação foi feita com o propósito de apurar os dados obtidos na entrevista e validar as regras de negócio a ter em conta na programação de horários. Com o uso da observação foi possível ter uma visão mais generalizada do processo de criação de horários, colhendo assim os dados que permitiram o estudo pormenorizado do problema.

O uso de simulações é devido ao período em que o trabalho decorreu (nos meados do ano lectivo), pois já havia horários definitivos nas duas escolas. Este processo sempre é feito no início do ano lectivo.

2.2 Metodologia de desenvolvimento

a) Paradigma de programação

O paradigma de programação usado, foi o Paradigma Orientado à Objectos, por permitir a construção de programas de computadores que retratam a vida real, permite também que os sistemas de computadores sejam portáteis, manuteníveis e reaproveitáveis e mais claros. Neste paradigma o sistema é conceptualizado como um conjunto de objectos que trabalham juntos para completar uma determinada tarefa e que cada objecto possui propriedades. Cada objecto é uma parte separada do sistema, interagindo com os outros objectos específicos através das suas propriedades, de forma muito bem controlada.

b) Linguagens de programação

Para a codificação da aplicação e dos procedimentos de consultas e armazenamento de dados na base de dados foram combinadas e usadas as linguagens java e SQL. A seguir é apresentada uma descrição de cada uma delas:

i. Java

Java é uma linguagem de programação orientada a objectos. Esta linguagem foi usada no desenvolvimento do sistema de elaboração de horários, isto é, foi usada no desenho dos

formulários usados no registo de entidades como professores, disciplinas, cursos, salas, turmas e no registo de relações existentes entre estas. Foi também usada no desenho da própria face da aplicação e na codificação dos algoritmos usados na elaboração de horários.

Por ser totalmente orientada a objectos, esta linguagem permite a herança e a reutilização de códigos, de forma dinâmica e estática, para além de ser a que o autor domina, ela tem ainda a vantagem de ser de código aberto e ter mais livros disponíveis, pois em caso de dúvidas poderiam ser consultados daí a razão da sua escolha para o desenvolvimento do sistema.

ii. SQL (Linguagem de Consulta Estruturada)

SQL é uma linguagem informática destinada a armazenar, manipular e obter dados armazenados em bases de dados relacionais. Esta linguagem foi usada na criação de tabelas, na codificação de procedimentos de consultas e armazenamento, remoção e actualização de dados na base de dados.

A linguagem SQL é adoptada por todos os Sistemas de Gestão de Bases de Dados (SGBD) relacionais existentes no mercado, além de ser uma linguagem padronizada pela American National Standards Institute (ANSI) (Instituto Nacional Americano de Padrões). Assim a mudança do SGBD não implicará a mudança do código nos procedimentos usados para manipular a base de dados. Pode ser utilizada tanto em máquinas Intel rodando Windows, passando por workstations rodando UNIX, até mainframes rodando sistemas operativos proprietários.

c) Sistema de Gestão de Base de Dados (SGBD)

Para o desenho da base de dados foi usado o MySQL como SGBD, pois é de código aberto, mais popular e desenvolvido e é muito fácil de se usar. Por ser de código aberto assim facilitará o uso e não haverá gastos em termos de valores monetários para a sua aquisição.

d) Técnicas de Modelação

Na modelação foi usada a linguagem *Unified Modelling Language* (UML) no desenho de diagramas de entidade e associações, de classes, de casos de uso, de sequência de eventos e de actividades.

Segundo Rumbaugh (2008) a UML é muito expressiva e abrange todas as visões necessárias ao desenvolvimento e implantação de sistemas:

- Visão de caso de uso: focaliza os comportamentos de um sistema devendo ser transparente a todos os envolvidos: gerentes, analistas, programadores e usuários finais.
 - Visão de Projecto: focaliza a estrutura de um sistema através da definição de classes, colaborações e as interfaces do sistema.
 - Visão de Processo: focaliza as questões de desempenho e escalabilidade do sistema.
 - Visão de Implementação: focaliza os artefactos físicos (programas, bibliotecas, banco de dados) para a efectiva montagem do sistema.
 - Visão de Implantação: focaliza a topologia do hardware, liberação e instalação do sistema.
- Um diagrama é a apresentação gráfica de um conjunto de elementos e são desenhados para permitir a visualização de um sistema sob diferentes perspectivas. A UML disponibiliza diagramas específicos para a modelagem visual das 5 visões:

- i. Diagrama de casos de uso para ilustrar as interacções do usuário com o sistema
- ii. Diagrama de classe para ilustrar a estrutura lógica

e) Ferramentas de desenvolvimento

Hoje em dia existem várias ferramentas, com interfaces amigáveis que permitem modelar e criar BD relacionais, bem como desenvolver aplicações. Para o desenvolvimento do Sistema de Elaboração de Horários, foram usadas ferramentas que a seguir são descritas:

i. Ferramentas usadas para o desenho da Base de Dados

Para a base de dados foi usada a ferramenta de desenho visual XAMP por ser muito fácil de se usar e permite o desenho de uma base de dados consistentes.

ii. Ferramentas usadas para o desenho da interface da aplicação

Onetbeans é para além de ser um compilador de java, um Framework genérico, gratuito e de código aberto, foi usado como editor de texto no desenho da interface gráfica e para a codificação dos algoritmos usados para a elaboração de horários, por ser de código aberto e por facilitar a programação pois sublinha logo os eventuais erros de sintaxe durante o desenvolvimento do sistema, para além de ajudar a identificar as propriedades (métodos) e atributos das classes.

Permite ainda a criação de um executável que foi usado no processo de instalação da aplicação;

Para além do netbeans foi usado o hexapixel, que é também um Framework de código aberto que fornece um menu rico em recursos e muito fácil de se usar e de se personalizar.

3. Fundamentação teórica

Neste capítulo de revisão de literatura apresento uma discussão sobre os trabalhos existentes na literatura de métodos computacionais, para produção de horários, com mais destaque os que tratam problemas similares ou relacionados ao de programação de horários escolares.

Um horário é basicamente uma tabela que descreve a alocação de recursos, no tempo para a realização de um conjunto de tarefas (Dicionário Editora de Língua Portuguesa, 2013). No caso em estudo, consideram-se recursos as salas, turmas e docentes. Por tarefas a realizar, entende-se o conjunto de aulas previstas, a leccionar em cada semana. Os recursos são alocados segundo, uma tabela de tempos lectivos que se distribui numa semana dividida em dias ou períodos de tempo e estes ainda em tempos unitários.

Guéret et al. (1994) define o problema de programação de horários escolares como uma listagem de certo número de prelecções, para serem frequentados por um grupo específico de estudantes e ministradas por um professor em um período definido de tempo. Cada prelecção requer certos recursos (salas e custos) em um número limitado e que certas restrições devem ser satisfeitas.

De acordo com o Burke E. et al (2006) as restrições podem ser geralmente divididas em dois tipos: restrições fortes e restrições fracas.

As restrições fortes têm de ser satisfeitas em qualquer circunstância. Por exemplo, em horários de aulas, um professor não pode ser alocado para ministrar aulas em duas turmas distintas simultaneamente. Uma solução que apresentar pelo menos uma das restrições fortes violada é considerada solução não exequível (não viável) e deve ser rejeitada.

As restrições fracas precisam de ser satisfeitas tanto quanto possível. Por exemplo em horários de aulas, as aulas de uma mesma disciplina, geralmente precisam de ser espalhados nos horários, de maneiras que não sejam ministradas em dias consecutivos.

Pela complexidade do problema de programação de horários, as restrições fracas podem precisar de ser relaxadas (violadas), uma vez que geralmente não é possível programar horários sem violar nenhuma delas.

Estas restrições de que o conjunto dos recursos necessários na programação de horários é sujeito, muitas vezes tornam o problema difícil de resolver em circunstâncias do mundo real

(Burke E. et al, 2006).

De acordo com o Bardadym (1996) a solução manual do problema de programação de horários escolares é uma tarefa árdua e normalmente requer vários dias de trabalho. Além do mais, a solução obtida pode ser insatisfatória com relação a vários aspectos. Por exemplo, em função da programação feita, pode haver dois ou mais professores alocados para ministrarem aulas no mesmo horário (tempo), na mesma turma e ou um professor ser alocado para ministrar aulas em mais de uma turma no mesmo tempo, prejudicando assim o processo do ensino e aprendizagem.

Em função de situações como estas, uma atenção especial vem sendo dada à automação do problema de programação de horários em escolas.

Existem três categorias a considerar na programação de horários escolares: Programação de horários de aulas, programação de horários dos exames e programação de horários de ocupação de salas, sendo que este trabalho ocupa-se à programação de horários de aulas.

Para fazer um horário escolar que atenda interesse de professores e dos recursos de uma instituição de ensino, deve-se considerar a especificidade de cada docente: disponibilidade de dias e horários e quais disciplinas ministra.

O problema de programação de horários vem afligindo matemáticos e pesquisadores da ciência há mais de 50 anos, pois segundo Lobo (2005), uma das primeiras referências para este problema é apresentada em (Appleby et al,1960). Appleby et al (1960) apresenta técnicas para construção de uma solução por intermédio de computadores, sendo feitas comparações entre problemas de quadro de horários e outros problemas de agendamento típicos da época.

Há diferentes algoritmos de otimização que têm sido usados para a programação automatizada de horários, Bertsimas & Tsitsiklis (1997) classificam estes algoritmos em três principais grupos:

a. Algoritmos Exactos

Algoritmos capazes de garantir a solução ótima. Geralmente desprendem muito tempo computacional e normalmente são inviáveis em aplicações reais. São exemplos desse grupo de algoritmos: *branch and bound*, *branch and cut*, algoritmo *Simplex* e programação dinâmica.

b. Algoritmos de Aproximação

São algoritmos capazes de encontrar soluções factíveis e que determinam o seu grau de optimalidade, ou seja, classificam quão boa é a solução. Geralmente, as soluções encontradas não são ótimas.

c. Heurísticas

Algoritmos capazes de encontrar soluções sem a garantia de exequibilidade ou optimalidade. Geralmente, desprendem pouco tempo computacional. São exemplos deste grupo de algoritmos: métodos de busca local, Busca Tabu, Simulated Annealing, Grasp e algoritmos evolutivos.

Este trabalho discute este grupo de algoritmos pois são os indicados na literatura para a resolução do problema de programação de horários ou típicos.

Métodos Heurísticos

Segundo Hertz (1987), heurísticas são técnicas usadas em situações que podem ser modeladas como problemas de maximizar ou minimizar uma função cujas variáveis têm certas restrições. Estas técnicas têm-se mostrado bastante eficientes por oferecerem melhores soluções e geralmente com tempo de processamento menor do que por outros tipos de técnicas.

A programação de horários pertence à classe de problemas NP-Difíceis. Por esta razão, em geral estes problemas são equacionados e resolvidos utilizando heurísticas em situações reais, pois dificilmente se encontraria algoritmos exactos capazes de retornar soluções para estes problemas em um tempo computacional razoável (Gendreau, 2003).

De forma geral, utilizam combinações de escolhas aleatórias e conhecimento histórico (dos resultados anteriores adquiridos pelo método) para se guiarem e realizarem suas buscas pelo espaço de pesquisa em vizinhanças, o que evita paradas prematuras em ótimos locais.

A seguir serão descritos métodos heurísticos propostos para a produção automatizada de horários para instituições educacionais.

3.1 Busca Local

Em um algoritmo de busca local define-se uma vizinhança composta por um conjunto de soluções com características próximas às de cada solução inicial.

Um algoritmo de busca local implementa-se a partir de uma solução corrente, em que percorre-se a vizinhança desta em busca de outra solução. Se tal solução vizinha for encontrada, torna-se a nova solução corrente e o algoritmo continua. Caso contrário, a solução corrente é um ótimo local em relação à vizinhança adoptada (Gendreau 2003).

A busca local é um procedimento interactivo que progressivamente melhora uma solução inicial factível, utilizando diversos operadores ou movimentos. O ponto ótimo de uma função é o ponto em que a função atinge seu ponto máximo, no entanto muitas funções possuem comportamento oscilatório apresentando picos de valores em determinada região (ver a figura 1). Se um ponto é um ponto máximo em determinada região diz-se que este ponto é um “Máximo Local”. Mas se o ponto é o ponto de maior valor em todo o domínio da função, é chamado de “Máximo Global” Albrecht (2005).

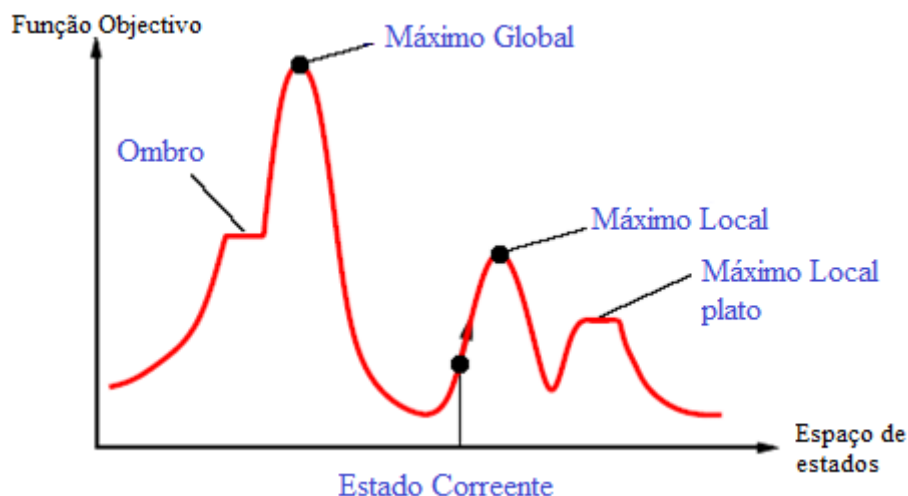


Figura 1 - Pontos da função de busca local (Fonte: Stuart Russel e Peter Norving (2007))

O sucesso de uma busca local está ligado directamente à definição de sua vizinhança. A vizinhança de uma solução consiste em todas as possíveis soluções que podem ser atingidas, a partir da aplicação dos operadores de busca disponíveis sobre outra solução. Operadores, ou movimentos, são pequenas perturbações que modificam alguma solução a fim de se atingir outra diferente. Com isso, a vizinhança e os tipos de operadores estão estreitamente ligados ao tipo de codificação utilizada para representar uma solução.

Ferland & Lavoie (1992) apresenta uma modelagem de programação inteira 0-1 para problemas de programação de horários. Nesse procedimento, a busca local primeiramente é aplicada para a diminuição progressiva das impossibilidades de execução de horários. Logo que uma solução factível é obtida, inicia-se a busca por melhores soluções.

Para a realização do método de busca local usam-se três etapas: na primeira utilizam-se trocas simples como busca local, o segundo procedimento realiza uma busca recursiva que apresenta similaridades em relação ao método de busca tabu e finalmente um procedimento de relaxamento lagrangeano

Segundo Faigle & Kern (1993) a busca local é relativamente genérica, permite a modelagem de várias restrições fortes mas não permite a modelagem de problemas com restrições como as existentes na programação de horários escolares com restrições de compacidade, isto é, horários que satisfaçam as necessidades pedagógicas e dos professores no que diz respeito à sua disponibilidade. A seguir apresenta-se o algoritmo do método busca local na tabela 3.

Algoritmo_Busca_Local Iterada

1. S_0 Gera_Solução_Inicial;
2. $S^* \longleftarrow$ Busca_Local(S_0);
3. Enquanto não é satisfeito um critério de parada,
 - 3.1 $S' \longleftarrow$ Perturbação(S^* , histórico);
 - 3.2 $S'^* \longleftarrow$ Busca_Local(S');
 - 3.3 $S^* \longleftarrow$ Critério_de_Aceitação(S^* , S'^* , histórico);
4. Fim-Enquanto;

3.2 Busca Tabu (BT)

Em 1986, Fred Glover (1986) propõe uma nova meta-heurística, que conta com uma estrutura de memória para guiar o algoritmo de busca local, conhecida como Busca Tabu. O termo meta-heurística desde então é comumente usado para designar métodos que guiam de forma inteligente uma heurística. Assim, BT é uma meta-heurística de busca local dotada de uma estrutura de memória que guarda informações das últimas soluções visitadas guiando a convergência do método pelo espaço de busca Hertz *et al.* (1995).

A BT é um método que evita retornar a um ótimo local previamente visitado de modo a para que se atinja um resultado ótimo global ou próximo, partindo de uma solução inicial (ver a figura 2)

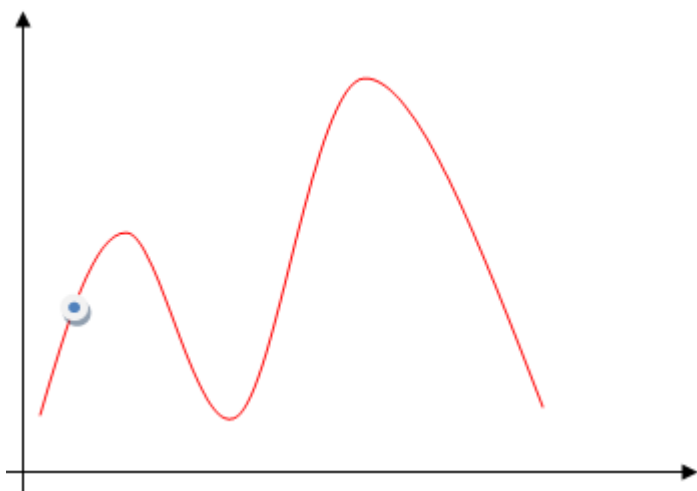


Figura 2 - Solução inicial (Adaptado de: Stuart Russel e Peter Norving (2007))

A busca move-se, a cada iteração, para a melhor solução na vizinhança conforme ilustra a figura 3, não aceitando movimentos que levem a soluções já visitadas por permanecerem armazenadas em uma lista denominada lista tabu.

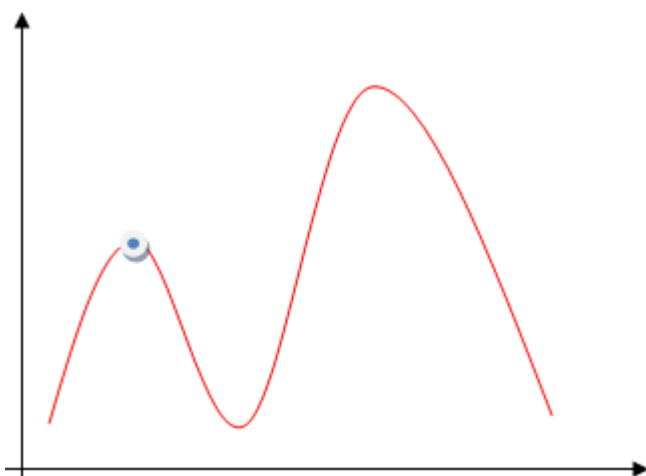


Figura 3 - Ótimo Local (Adaptado de: Stuart Russel e Peter Norving (2007))

A lista permanece na memória guardando soluções já visitadas (tabu) durante um determinado tempo (ver a figura 4). No fim espera-se que a busca tenha encontrado um resultado ótimo global ou próximo a este (ver a figura 5).

A estrutura de memória da BT é um factor importante no caminho por onde a busca é conduzida, pois estruturas de memória muito grandes diversificam a busca, pois os movimentos ditos promissores são proibidos durante muitas iterações. Estruturas de memória muito pequenas intensificam a busca, pois os movimentos promissores são liberados mais rapidamente (Glover, 1989).

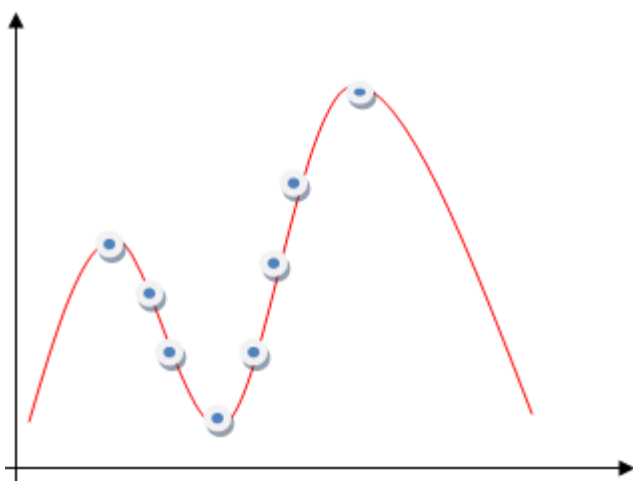


Figura 4 - Elementos Tabu (Adaptado de: Stuart Russel e Peter Norving (2007))

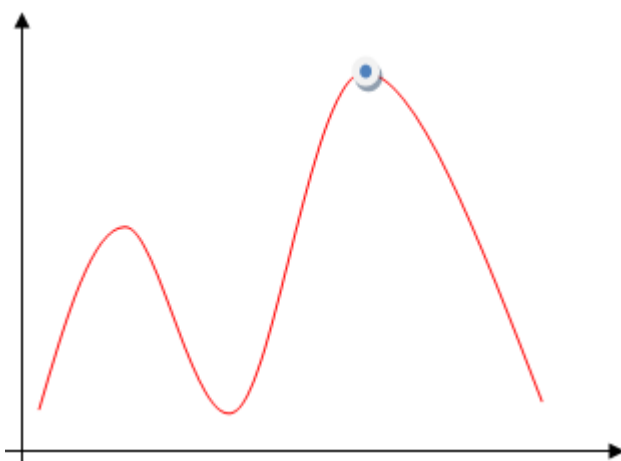


Figura 5 - Ótimo Global. (Adaptado de: Stuart Russel e Peter Norving (2007))

Glover e Laguna (1997) estudaram o impacto destas variações. Também avaliaram o uso de técnicas de intensificação em locais promissores e diversificação da solução quando a busca fica presa em mínimos locais. Laguna (1995) descreve o impacto da busca exaustiva em problemas de menor e maior complexidade.

Lu & Hao (2010) utilizam uma versão adaptativa da BT para solucionar um problema de programação de horários com penalizações que guiam os operadores e sistema de intensificação e diversificação do espaço de busca.

Hertz (1987) propôs métodos de busca tabu para a geração de horários em universidades baseados em uma implementação prévia do método para coloração de grafos. Uma particularidade dessa proposta é a decisão de colocar a violação de algumas restrições fracas fora do espaço de busca e tentar então diminuir a ocorrência de conflitos. Segundo Albrecht (2005), o teste de busca tabu em duas instituições mostrou que as soluções deste método são consideravelmente melhores do que as soluções manuais.

Mais especificamente tratando de problemas de programação de horários em escolas, algoritmos baseados em busca tabu foram propostos por (Costa, 1994; Schaerf, 1999 e Alvarez-valdés, 1996). Na busca tabu de Costa a vizinhança é definida pela mudança no período de alocação de uma determinada aula.

A memória de curto prazo é implementada através de duas listas tabus: L1 e L2 que consideram, respectivamente, a movimentação de uma aula e a movimentação de uma aula para um

determinado período, sendo $L1$ menor que $L2$. O critério de aspiração considera alterações não somente no valor da função objectivo, mas em componentes desta. A diversificação é implementada através da oscilação dos pesos na função objectivo relacionados à ocorrência de conflitos: faz-se uma redução periódica desses pesos permitindo que o processo de busca visite outras regiões do espaço de busca.

O método híbrido baseado em Busca Tabu e Greedy Randomized Adaptive Search Procedure

(GRASP) proposto por Souza(2003), destacou-se entre os algoritmos híbridos estudados por (Souza, 2000), onde são apresentadas e avaliadas computacionalmente várias propostas de heurísticas para o problema de programação de horários escolares com restrições, incluindo Simulated Annealing, Busca Tabu e Optimização Microcanônica Torreão (1995). Os algoritmos propostos agregavam um método de melhoramento baseado em caminhos mínimos propostos por Souza para a pesquisa de movimentos compostos, o qual é utilizado tanto para acelerar a obtenção de soluções iniciais quanto para o melhoramento das mesmas.

Apesar de a Busca Tabu evitar ciclos, explorar zonas promissoras ou forçar a visita a novas zonas do espaço de procura e ser económico, pois tende rapidamente para um valor perto do óptimo, a procura não é completa, pois segundo Cemim (2005) o algoritmo tende a um óptimo local, mas não garante que se encontre a solução óptima e a procura não é discriminatória, ou seja, a solução encontrada depende do ponto de partida para a optimização.

De acordo com Ferro & Chui (2010) um grande problema da Busca Tabu é que, se o espaço de busca for muito grande, e particularmente, se for de alta dimensionalidade, a busca poderá ficar presa em uma mesma vizinhança, mesmo que utilize uma extensa Lista Tabu - com um imenso espaço de busca, pode simplesmente existir um excesso de posições para serem avaliadas. Como abordagem alternativa, pode se criar uma Lista Tabu não com as soluções candidatas, mas com as alterações recentemente efectuadas ao gerar vizinhos de uma solução.

A seguir é mostrado o algoritmo da Busca Tabu na tabela 4.

Tabela 4 - Procedimento Busca Tabu (Fonte: Costa F.P.(2003))

Algoritmo Busca_Tabu

1. Seja S_0 solução inicial;
2. $S' \leftarrow S$ {Melhor solução obtida até então}
3. $Iter \leftarrow 0$; {Contador do número de iterações}
4. $MelhorIter \leftarrow 0$; {Iteração mais recente que forneceu S' }
5. Seja $BTmax$ o número máximo de iterações sem melhora em S' ;
6. $T \leftarrow \{\}$; {Lista Tabu}
7. Inicialize a função de aspiração A ;
8. Enquanto ($Iter - MelhorIter \geq BTmax$) Faça
9. $Iter \leftarrow Iter + 1$;
10. Seja $S' \leftarrow S$ m o melhor elemento de $V \setminus N(S)$ tal que o movimento m não seja tabu
11. ($m \in T$) ou S' atenda a condição de aspiração ($f(s') < A(f(s))$);
12. Actualize a Lista Tabu T ;
13. $S \leftarrow S'$;
14. Se $f(s) < f(s')$ então
15. $S' \leftarrow S$;
16. $MelhorIter \leftarrow Iter$;
17. Fim-se
18. Actualize a função de aspiração A ;
19. Fim-enquanto;
20. Retorne s' ;
21. Fim Busca_Tabu;

3.3 Algoritmos Evolutivos

Holland (1975) propôs uma nova heurística baseada na evolução das espécies de Darwin, chamada Algoritmo Genético (AG). O AG é um algoritmo evolucionário que, através da geração de uma população de indivíduos (soluções), da troca de genes (recombinações) e de mutações,

realiza a pesquisa no espaço de busca. Todavia, o AG se popularizar mais tarde com o trabalho desenvolvido por Goldberg (1989).

A estrutura básica do algoritmo genético é composta por: geração da população inicial, avaliação de aptidão, os operadores de selecção, cruzamento e mutação e uma condição de parada, conforme apresentado na figura 6.

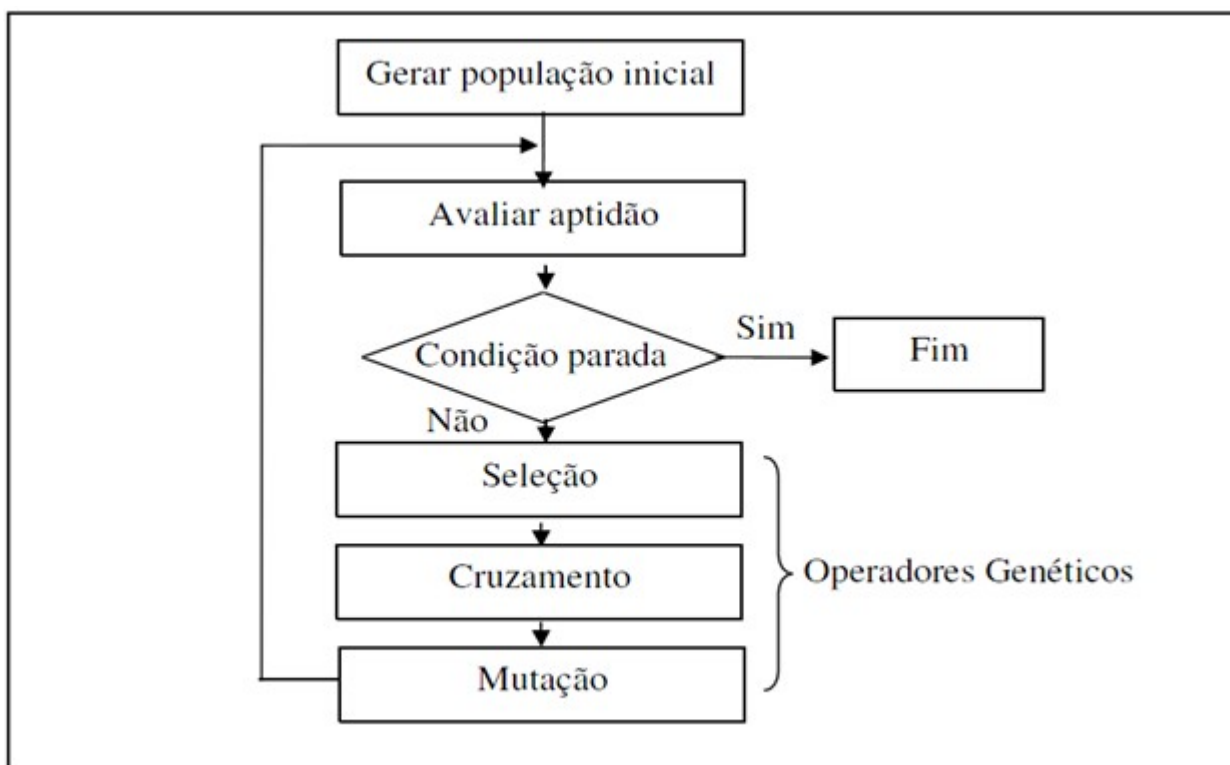


Figura 6 - Pseudocódigo do algoritmo evolutivo (Fonte: Benz (2005))

A população de um AG pode muitas vezes convergir rapidamente para mínimos locais, tendo em vista que soluções que caracterizam este ponto de mínimo espalham seus genes mais rapidamente devido ao seu valor de aptidão.

Vários algoritmos genéticos foram propostos para a resolução de problemas de programação de horários. Abramson & Abela, (1992) propuseram um algoritmo genético paralelo para a resolução de problemas de programação de horários em escolas. No problema considerado, somente leva-se em conta a não existência de conflitos. O algoritmo genético híbrido de Colorni, Dorigo e Maniezzo (2000) considera um problema com mais requisições reais. Nesse algoritmo a combinação de operadores genéticos que consideram características do problema e uma busca

local em múltiplas vizinhanças é utilizada. Comparações com implementações bastante simples de Simulated Annealing e Busca Tabu são apresentadas, sendo que os melhores resultados foram apresentados pelo algoritmo genético híbrido e pela busca tabu.

3.4 Métodos Baseados em Programação Linear

Técnicas de Programação Linear, incluindo Programação Linear Inteira, também têm sido utilizadas para a geração de quadros de horários. Na maioria dos casos, busca-se a produção de quadros de horários ótimos.

Lawrie (1996), apresenta uma aplicação que considera o problema de não ocorrência de conflitos, pois para tal cabe ao usuário do software informar previamente quais cursos podem ser oferecidos simultaneamente, através da criação de arranjos.

Mais recentemente, houve avanços na capacidade de resolução de problemas de Programação Linear Inteira Mista que permitiram a resolução de problemas de programação de horários nos trabalhos de (Birbas et al, 1997; Daskalaki, 2004; Avella & Vasil'ev, 2005). Birbas et al (1997) por exemplo, especifica um conjunto de dias de folga para cada professor, já Avella, & Vasil'ev (2005) especifica um número máximo de dias em que o professor deve comparecer à instituição de ensino. Essa abordagem, apesar de possibilitar uma considerável diminuição na dimensão do espaço de busca, traz a desvantagem de que na especificação do número de dias de folga ou trabalho de cada professor devem ser feitas escolhas sábias: a requisição de horários tão compactos quanto possível considerando cada professor isoladamente pode resultar em um problema sem solução viável. A alteração manual do modelo para permitir a exequibilidade e oferecer quadros de horários satisfatórios para professores pode excluir soluções ótimas quanto a esse requisito.

Papoutsis et al (2003) apresenta uma aplicação de Geração de Colunas para um modelo com um número exponencial de variáveis para uma variante do problema de programação de horários encontrada na Grécia é apresentada. Nesse modelo, cada variável binária representa um possível quadro de horários semanal de um professor, sendo que a maioria das restrições é considerada na geração das colunas. O procedimento continua iterativamente com a adição de novas colunas

enquanto alguma melhoria significativa na função objectivo é observada. Uma vez que a solução resultante do procedimento apresentado geralmente não é inteira, os autores desenvolveram um procedimento específico que tenta tornar factível a solução. O procedimento por esses autores é heurístico, sem garantia de produção da solução óptima.

3.5 Recozimento Simulado (*Simulated Annealing*)

Em 1983, uma impressionante técnica de busca local foi descrita, o *Simulated Annealing* (SA) de Kirkpatrick *et al.* (1983). Tratava-se de um método baseado no sistema de resfriamento dos corpos e que tinha convergência para o óptimo global garantida quando o número de iterações da busca tendia ao infinito.

Segundo Albrecht (2005) SA é um método de busca local que aceita soluções de pior qualidade baseada em uma curva logarítmica controlada por um parâmetro chamado temperatura. A temperatura está directamente ligada à probabilidade de aceitação de soluções de pior qualidade. Geralmente, no início do procedimento, a temperatura possui um valor alto, e este valor diminui com o decorrer das iterações de acordo com uma função de resfriamento, caracterizando assim a convergência da busca.

Este método desde então se popularizou bastante pela facilidade de implementação e pelos bons resultados alcançados.

Jin *et al.* (2009) apresenta uma aplicação de *simulated annealing* em um problema de programação de máquinas simples baseados em uma indústria de fabricação de correntes da China. As definições clássicas de operadores de busca (como o movimento de troca simples, por exemplo) não surtiriam efeito neste problema tendo em vista a existência de famílias de produtos. Isso significa que produtos da mesma família não apresentam tempo de troca de um produto para o outro. Dessa forma, Jin *et al.* (2009) desenvolvem novos operadores de busca baseados na destruição e construção de famílias de produtos, onde utilizam um procedimento guloso na montagem de novos lotes conseguindo resultados melhores que outros métodos da literatura.

4. Estudo de Caso

O presente trabalho foi realizado nas Escolas Primária Completa 1 de Junho e Secundária da Zona Verde. A razão da escolha destes dois níveis prende-se com o facto de horários no secundário apresentarem janelas (furos), situação que não se verifica nos horários das escolas primárias. Portanto o envolvimento das duas escolas, permitiu o estudo pormenorizado do problema.

Constitui ainda a razão da escolha destes, o facto de o autor ter estudado na Escola Secundária da Zona Verde e trabalhado na Escola Primária Completa 1 de Junho, onde acompanhou por perto as dificuldades enfrentadas no processo de elaboração de horários escolares.

Por se localizarem na zona residencial do autor, tornou-se fácil a deslocação e a colheita de dados.

a) Escola Primária Completa 1 de Junho

A Escola Primária Completa 1 de Junho é uma instituição de ensino primário localizada na cidade de Maputo, no Distrito Municipal Kamubukwana, no Bairro George Dimitrov. Possui uma estrutura física composta por quinze (15) salas de aulas e um edifício onde funcionam os sectores administrativo e directivo.

Quanto à estrutura organizacional, é composta por uma Direcção Pedagógica e uma Direcção administrativa. Esta última é responsável por assegurar, praticamente, a locação e a gestão de recursos humanos, físicos e financeiros.

A Direcção Pedagógica é responsável pela coordenação do exercício docente de cerca de 41 professores, distribuídos por ciclos e por classes.

A Escola Primária completa 1 de Junho lecciona dois ciclos do Ensino primário denominados:

Primeiro (EP1) e Segundo (EP2) ciclos. O primeiro ciclo comporta desde a 1ª até 5ª classe, distribuídas por três turnos diurnos conforme mostra a tabela 5, cada turno com uma média de 14 turmas.

Tabela 5 - Distribuição das classes de EP1 por turno da Escola Primária Completa 1 de Junho

Turno	Classes	Entrada	Saída
Primeiro	3ª e 5ª	6h:30min	10h:15min
Segundo	1ª e 2ª	10h:20min	14h:05min
Terceiro	4ª	14h:10min	17h:55min

Fazem parte do segundo ciclo a 6ª e 7ª classes. As aulas deste ciclo decorrem em dois turnos: o primeiro e o segundo. A tabela 6 mostra a distribuição das classes do segundo ciclo, cada um com uma média de 8 turmas.

Tabela 6 - Distribuição das classes de EP2 por turno da Escola Primária Completa 1 de Junho

Turno	Classe	Entrada	Saída
Primeiro	6ª	6h:30min	11h:00min
Segundo	7ª	13h:30min	17h:55min

b) Escola Secundária da Zona Verde (ESZV)

Esta escola está localizada no Município da Matola, no Bairro da Zona Verde. Possui uma estrutura física composta por vinte (20) salas de aulas, um edifício onde funciona a Direcção desta, dois (2) laboratórios, uma sala de informática, uma biblioteca, um centro social, um ginásio

multi-uso, um parque de estacionamento, uma sala de corte e costura e um canto de aconselhamento da Geração Biz. A ESZV lecciona também em dois ciclos do ensino secundário:

O primeiro ciclo e o segundo ciclos. O primeiro ciclo engloba deste a 8^a à 10^a classe. Já o segundo é constituído por apenas duas classes: 11^a e 12^a classes.

Possui uma estrutura organizacional formada por duas Direcções: administrativa e a Pedagógica, esta última com 60 professores distribuídos em cinco classes (ver tabela 7), estas por sua vez distribuídas em três turnos, sendo dois diurnos e o terceiro nocturno. As classes estão distribuídas segundo consta na tabela 7.

Tabela 7 - Distribuição das classes por turno da ESZV

Turno	Classes	Entrada	Saída
Primeiro	10 ^a , 11 ^a e 12 ^a	07h:00min	12h:30min
Segundo	8 ^a , 9 ^a e 10 ^a	13h:00min	17h:30min
Terceiro (Nocturno)	Todas	18h:00min	22h:30min

4.1 Situação actual

A elaboração de horários nestas escolas é actualmente feita manualmente, tarefa considerada árdua por exigir muita atenção e consumo de tempo medido em semanas e que frequentemente resulta em soluções insatisfatórias.

O sistema actual de elaboração de horários nestas escolas exige os seguintes requisitos:

Régua ou esquadro, cartolinas, canetas, lápis marcadores, códigos das disciplinas fornecidos pelos directores pedagógicos, giz e apagador e no mínimo três professores e uma lista das disciplinas da classe e/ou da secção. Esta lista deve ter os detalhes de cada disciplina, tais como:

o nome da disciplina, o código e a sua carga horária semanal. É preciso, também, conhecer o número de turmas por classe ou secção.

Para a elaboração de horários usa-se o seguinte procedimento:

- Desenharam-se nas cartolinas e no quadro as tabelas que formarão os horários das turmas assim como os dos professores;
- Um professor fica no quadro a fazer combinações nas tabelas enquanto os outros ajudam a controlar e a lançar nas cartolinas;
- Uma disciplina é seleccionada aleatoriamente, insere-se o código desta, em um número de células correspondente à carga horária da mesma disciplina, enquanto preenche-se as tabelas, verifica-se se as restrições são respeitadas em cada tabela. As inserções são feitas até que se completem as tabelas que correspondem aos horários das turmas. Do mesmo modo é seleccionada outra até que todas as disciplinas da classe sejam seleccionadas. No preenchimento das tabelas deve-se respeitar a carga horária de cada disciplina, isto é nenhuma tabela deve ter um número de aulas de certa disciplina inferior ou superior à sua carga horária.

As disciplinas com carga horária igual ou superior a três (3) devem ter blocos, isto é, as suas aulas devem ser dadas em duplas sendo uma dada singularmente se a sua carga horária for ímpar. No início, o processo parece ser simples, mas à medida que as células das tabelas vão se preenchendo, a complexidade aumenta, pois as possibilidades de preenchimento sem choques vão se reduzindo.

Segundo os programadores de horários destas escolas, é muito frequente começar-se o preenchimento de tabelas e quase no fim descobrir-se que a solução é insatisfatória e inviável. Quando isto acontece recomeça-se o processo.

É frequente também verificar-se a penetrabilidade, isto é, professores calharem na mesma turma simultaneamente. Quando assim acontece tenta-se acertar o horário; se não for possível, recomeça-se com o processo.

Depois da elaboração segue-se a divulgação e/ou distribuição dos horários.

A Escola Primária Completa 1 de Junho cria tabelas no computador em que serão copiados os dados das tabelas nas cartolinas, para a sua posterior impressão. Já a Escola secundária da Zona Verde cria tabelas manualmente no papel A4. Depois da criação das tabelas, faz-se o recorte das mesmas para serem distribuídos aos professores, assim como às turmas. Os horários das turmas são também colados nas vitrinas.

Com base no acima exposto, no que diz respeito à elaboração de horários, inferimos que as Tecnologias de Informação e Comunicação só são usadas pela Escola Primária 1 de Junho na divulgação de horários.

A maior vantagem do sistema actual de elaboração de horários (manual) reside no facto de poder correr com ou sem recursos como computadores e impressoras, para além de apresentar poucos gastos. Porém, apresenta desvantagens penosas, tais como:

- Maior ciclo de vida das suas actividades, isto é, consome-se muito tempo na elaboração, tempo este quantificado em dias.
- Apresenta muitas vezes resultados insatisfatórios o que compromete a qualidade do ensino;

5. Análise e discussão de dados

O presente capítulo trata da descrição de dados e discussão de resultados, colectados a partir da pesquisa documental e entrevistas. Foi feita uma entrevista de 23 perguntas a um universo de 20 professores das Escolas Primária Completa 1 de Junho e da Escola Secundária da Zona Verde, com a finalidade de obter mais informação acerca do problema da montagem das grades horárias escolares, é de salientar que os professores entrevistados são os que lidam com a elaboração de horários escolares para gerir o tempo lectivo.

Relativamente à primeira e à segunda perguntas, sobre o número de turnos e horários aplicados nestes, na Escola Primária Completa 1 de Junho as aulas são ministradas em três turnos distintos (primeiro, segundo e terceiro) no EP1, e no EP2 em dois turnos (primeiro e segundo), todos diurnos. Já na Escola Secundária da Zona Verde as aulas são dadas só em três turnos (diurno - manhã, diurno – tarde e nocturno). A seguir na tabela 8, são apresentados horários dos diferentes turnos aplicados nas duas escolas.

Tabela 8 - Distribuição de horários nos diferentes turnos

	Turno	Entrada		Saída	
Escola Primária					

A questão três trata das estruturas organizacionais das escolas, ambas possuem sectores pedagógico e administrativo.

Na EPC1 de Junho o sector pedagógico coordena o exercício docente de cerca de 41 professores e na ESZV o sector pedagógico apresenta o corpo docente de cerca de 60 professores. Como se pode ver, cada professor deve ter o seu horário, o que significa que na EPC 1 de Junho são necessários 41 horários e na ESZV 60 correspondente ao número de professores, pelo elevado número de horários necessários fica claro que esta actividade não precisa um tempo considerável para o seu término.

No que tange às classes e a sua distribuição por turno que é o propósito das perguntas 6 e 7, A EPC 1 de Junho lecciona de 1ª a 7ª classes, distribuídos pelos três turnos. A ESZV lecciona de 8ª

a 12ª classes também distribuídos pelos três turnos. A seguir é apresentada a tabela 9 referente à distribuição de classes por turno.

Tabela 9 - Distribuição das classes por turno

Escola Primária		
Escola Secundária da		

Quando perguntados acerca do número de turmas A EPC 1 de Junho tem 43 turmas todas com salas fixas. A ESZV tem 59 turmas distribuídas pelos três turnos. Apesar de estas terem salas fixas elas partilham a sala de informática, o laboratório e o ginásio.

Quanto a duração da aula, na EPC 1 de Junho a aula dura 40 minutos e o intervalo de lanche dura 10 minutos e na ESZV a aula dura 45 minutos e o intervalo de lanche dura 15 minutos.

Em ambas o intervalo tem a duração de 5 minutos. Assim nota-se que os horários do ensino secundário são longos em relação aos do ensino primário.

Quando perguntados acerca da carga horária, esta varia de acordo com a secção conforme mostram as tabelas 10 e 11 a seguir apresentadas.

Tabela 10 – Relação de cargas horárias do ensino secundário

Total de carga horária semanal	28	29	28

Tabela 11 – Relação de cargas horárias do ensino primário

Total de Carga horária	30

Conforme mostram as tabelas 9 e 10, as cargas horárias totais do ensino secundário são inferiores em relação às do ensino primário. Assim pode-se inferir que, os horários das turmas do ensino secundário apresentam janelas (furos), enquanto os do ensino primário apresentam-se totalmente preenchidos, pois são 5 dias úteis por semana e por cada dia são geridos seis horários (tempos).

Desta forma tem-se $5 \times 6 = 30$ horários (tempos) semanais que correspondem ao total da carga horária semanal do ensino primário. Portanto é mais difícil construir uma grade horária numa escola primária do que numa secundária, pois em (Abramson, 1991) evidencia-se que maior número de janelas mais facilidade de construir a grade horária.

No que concerne às aulas geminadas, em ambas as escolas só é permitido um máximo de duas aulas por dia.

Ao serem interrogados acerca do tempo que dura a elaboração de horários, 25% afirmam que demora duas semanas, 65% afirmam que demora 3 semanas e 10% afirmam que demora 4 semanas. A figura 7 apresenta o gráfico relativamente ao tempo necessário para a elaboração de horários.

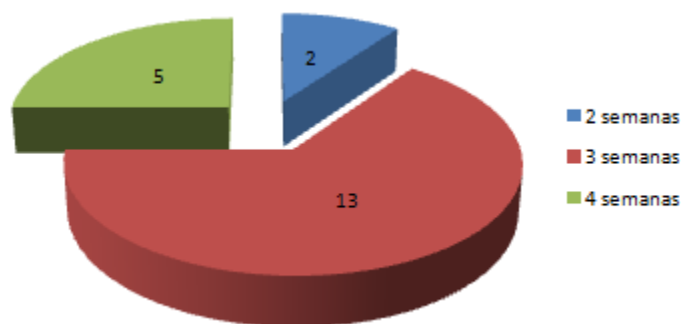


Figura 7 – Duração da elaboração de horários

Com base no gráfico nota-se que a elaboração de horários dura em média 3 semanas, tempo

No que diz respeito aos requisitos necessários para a elaboração de horários, ambas escolas afirmam que é necessário o seguinte: Régua, cartolinas, canetas, lápis marcadores, códigos das disciplinas, giz e apagador e no mínimo três professores e uma lista de códigos as cargas horárias das disciplinas de cada classe e/ou da secção. É preciso, também, conhecer o número de turmas por classe ou secção.

Quanto à indisponibilidade do professor, esses casos não se verificam, uma vez que o professor tem de estar em pelo menos um dos turnos a ser indicado pela direcção pedagógica. Pela complexidade da elaboração de horários, o difícil é contornar os casos de preferências dos professores, estes acabam sendo alocados aos horários aos quais não são da sua preferência.

Quanto ao grau de dificuldade 85% afirmam que a elaboração de horários é uma actividade complexa e difícil de se executar por estar repleto de restrições. 15% dos professores afirmam que a elaboração é uma actividade não difícil mas que precisa de concentração, paciência e tempo para a sua execução. Na figura 8 está o gráfico que ilustra o grau de dificuldade.

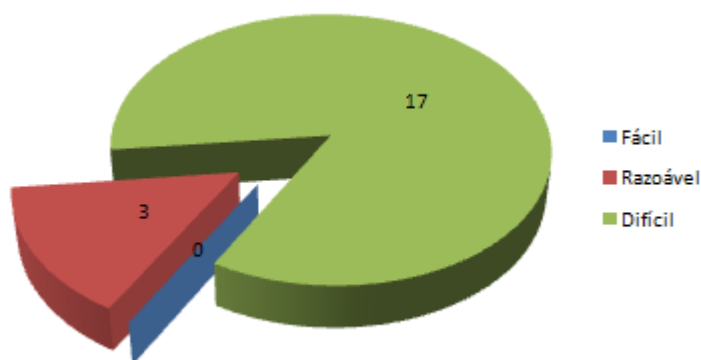


Figura 8 – Grau de dificuldade de elaboração de horários

No que concerne ao grau de satisfação, 30% afirmam estar satisfeitos com os horários produzidos, 55% afirmam que adaptam-se aos horários produzidos e 15% afirmam estar insatisfeitos. A figura 9 ilustra o grau de satisfação dos professores no que diz respeito aos horários produzidos pelas escolas

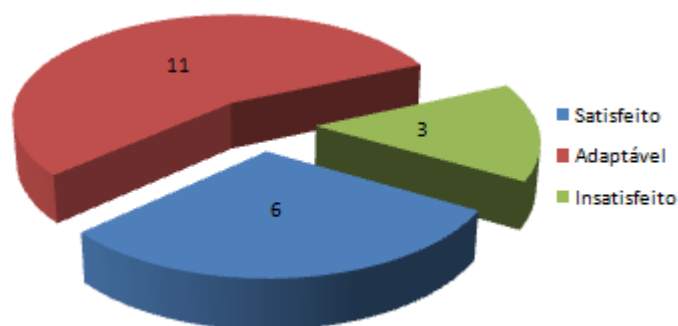


Figura 9 – Satisfação de professores pelos horários

5.1 Modelo Proposto

Com vista a conceber o software de geração de horários para minimizar os constrangimentos e problemas encontrados no processo de criação de horários escolares, neste capítulo são apresentados e descritos os diagramas de classes e diagramas de sequência de eventos, assim como os casos de uso envolvidos neste processo.

A Figura 10 apresenta o diagrama de classes para o sistema desenvolvido. Neste são apresentadas todas as entidades envolvidas no processo de elaboração de horários.

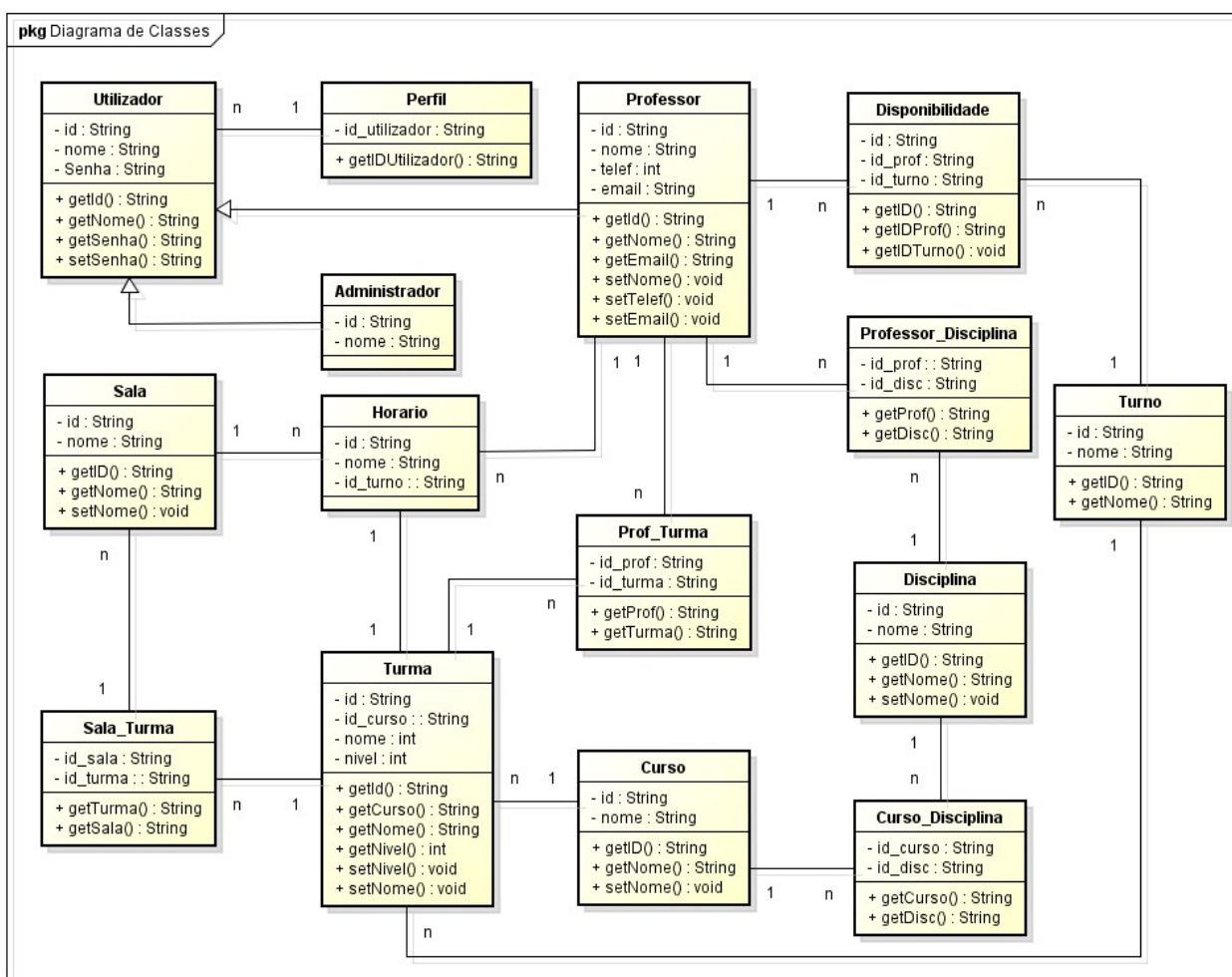


Figura 10 - Diagrama de classes

A seguir é apresentada a descrição dos actores do sistema, os quais representam funções desempenhadas por pessoas, quando interagem com o sistema.

A figura 11 representa dois casos de uso: O do Administrador do sistema e o do professor

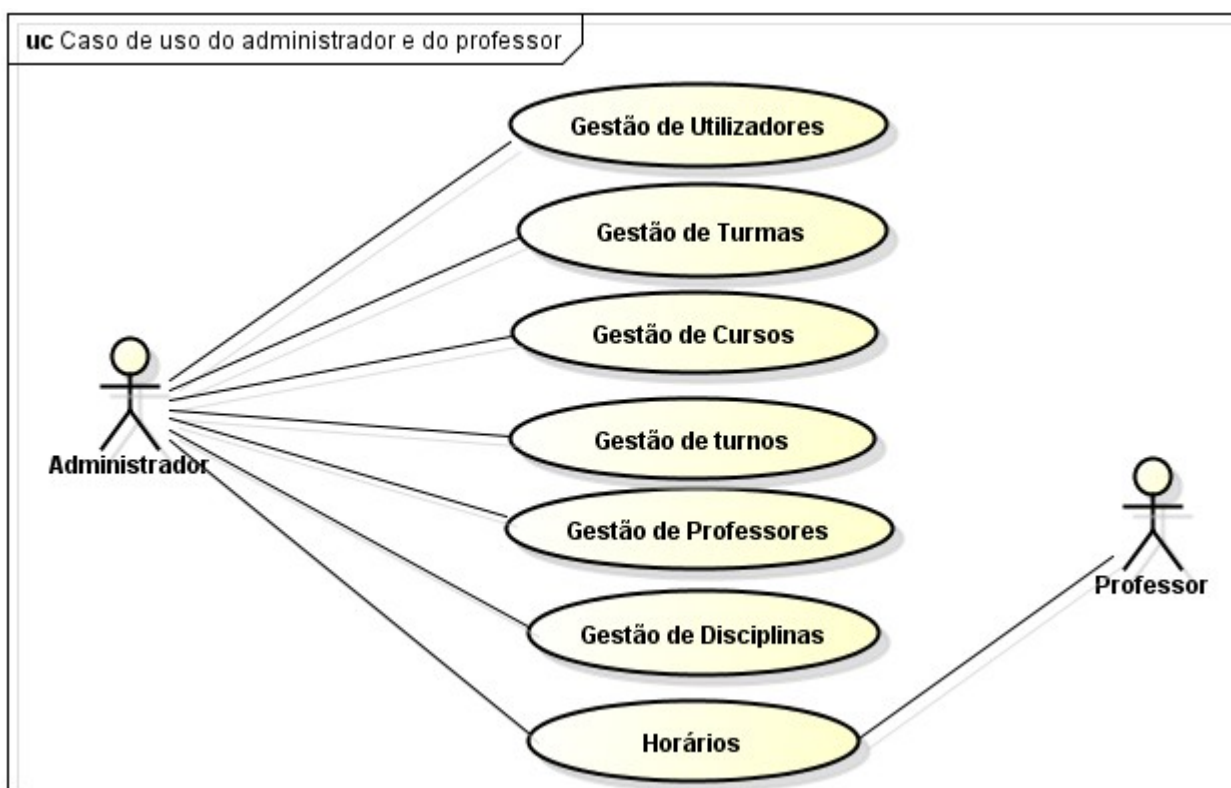


Figura 11 – Caso de uso do administrador e do professor

Nos sistemas de gestão, há sempre necessidade de controlar o acesso ao sistema. Este controle consegue-se através da gestão de utilizadores, em que é feito o mapeamento de privilégios e de perfis destes, de maneiras a se saber que operações cada um deles pode efectuar no sistema. A figura 12 mostra o caso de uso de gestão de utilizadores.

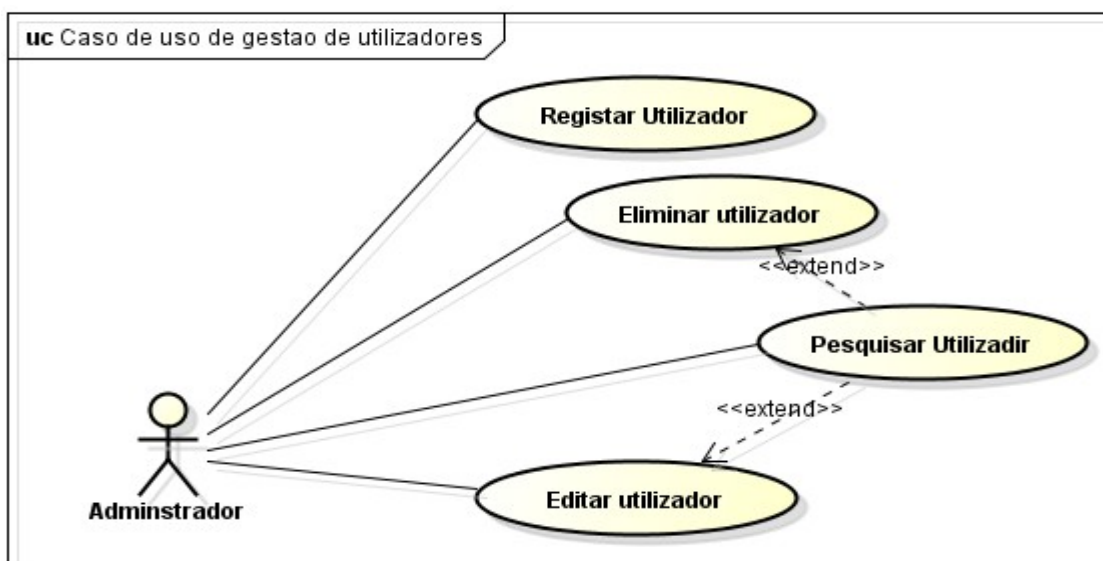


Figura 12 – Caso de uso de gestão de utilizadores

Os cursos e as disciplinas são algumas das entidades envolvidas na programação de horários escolares. Na figura 13, são apresentados os casos de uso de gestão do curso e da disciplina

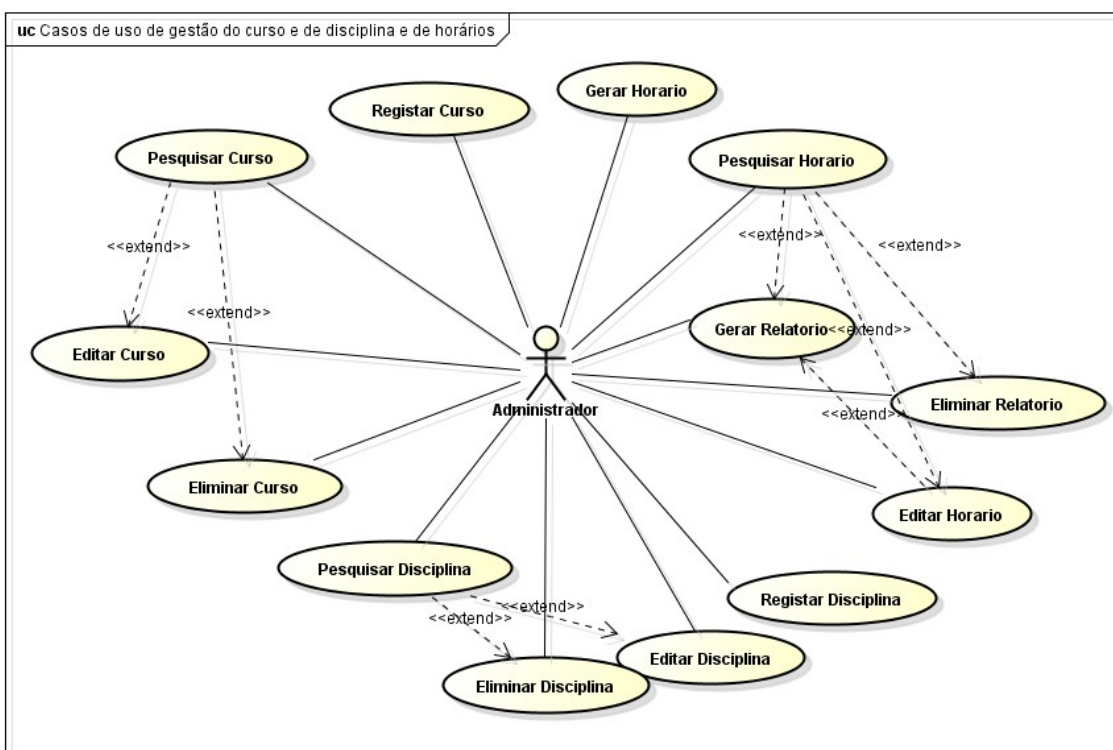


Figura 13 – Casos de uso de gestão de cursos e de disciplinas

Sendo a programação de horários escolares feita para professores e turmas, é imprescindível a gestão destas entidades incluindo o turno em que decorrerão as suas aulas. A seguir passamos a apresentar na figura 14, os casos de uso de gestão de turmas, de professores e de turnos.

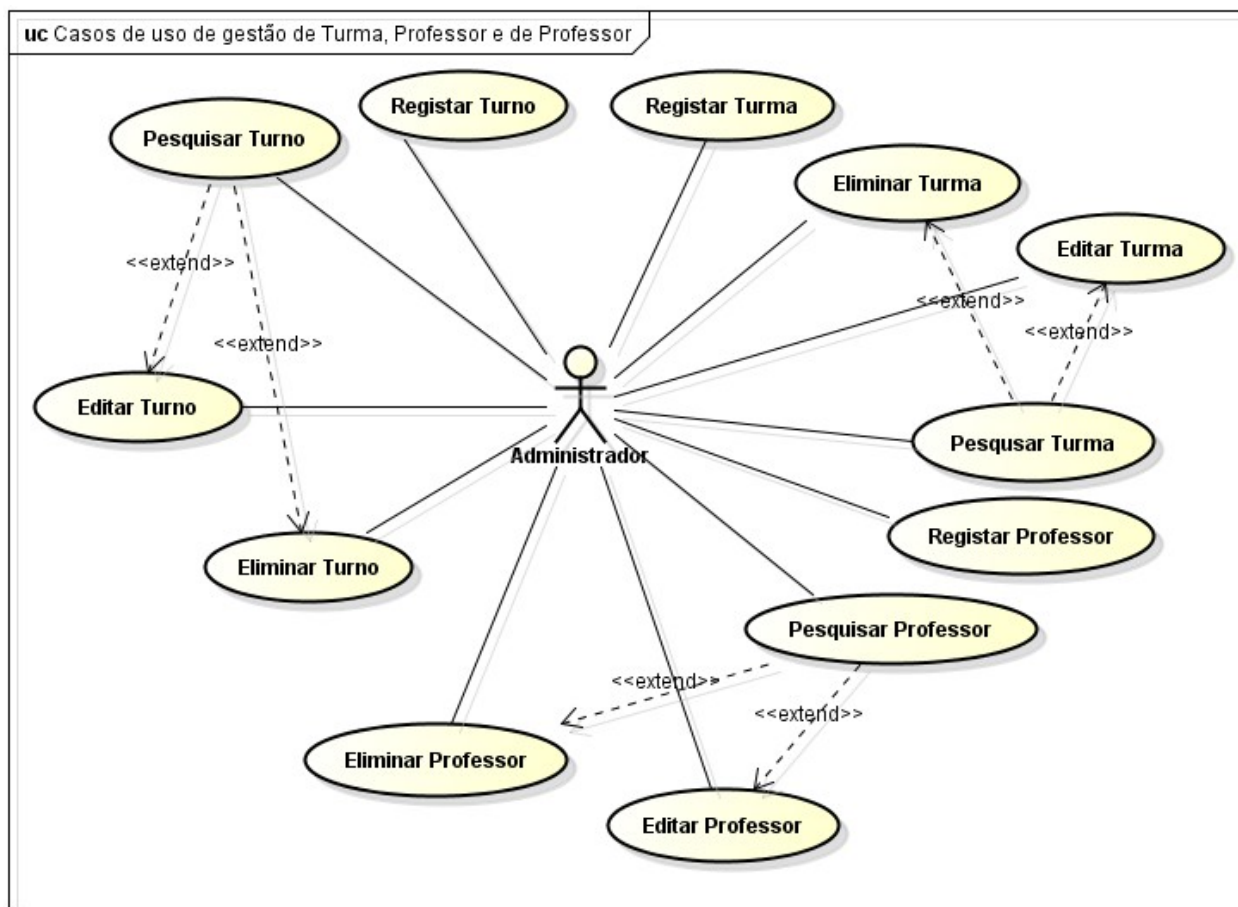


Figura 14 - Casos de uso

A seguir apresentamos os diagramas de sequência de eventos dos casos de uso acima apresentados. A figura 15 ilustra um conjunto de eventos desencadeados para gestão de disciplinas

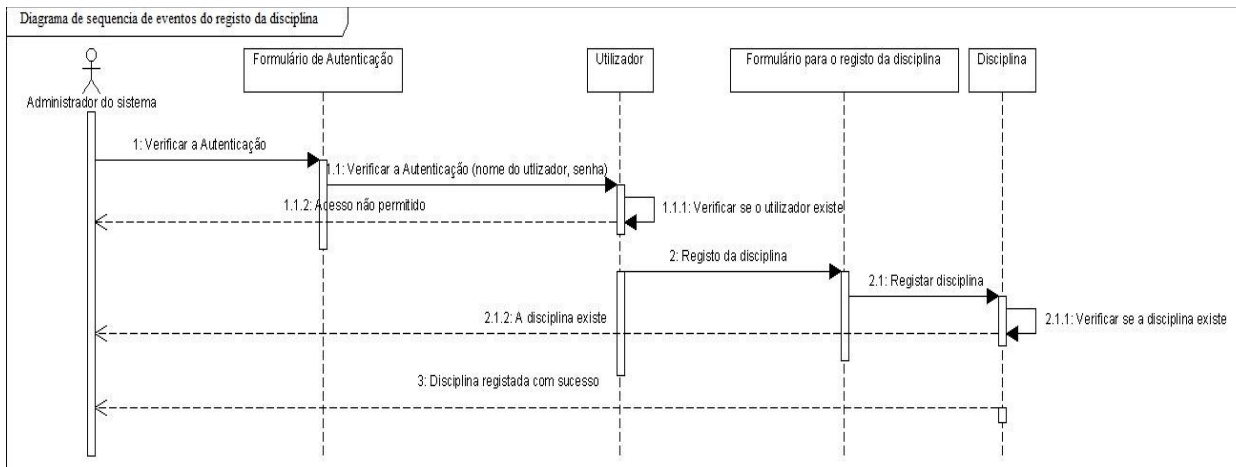


Figura 15 - Diagrama de sequência de eventos para registar a disciplina

Na gestão de cursos há um conjunto de eventos sequenciados que ocorrem, de modo a torná-la possível. A figura 16 ilustra a sequência destes eventos no registo do curso

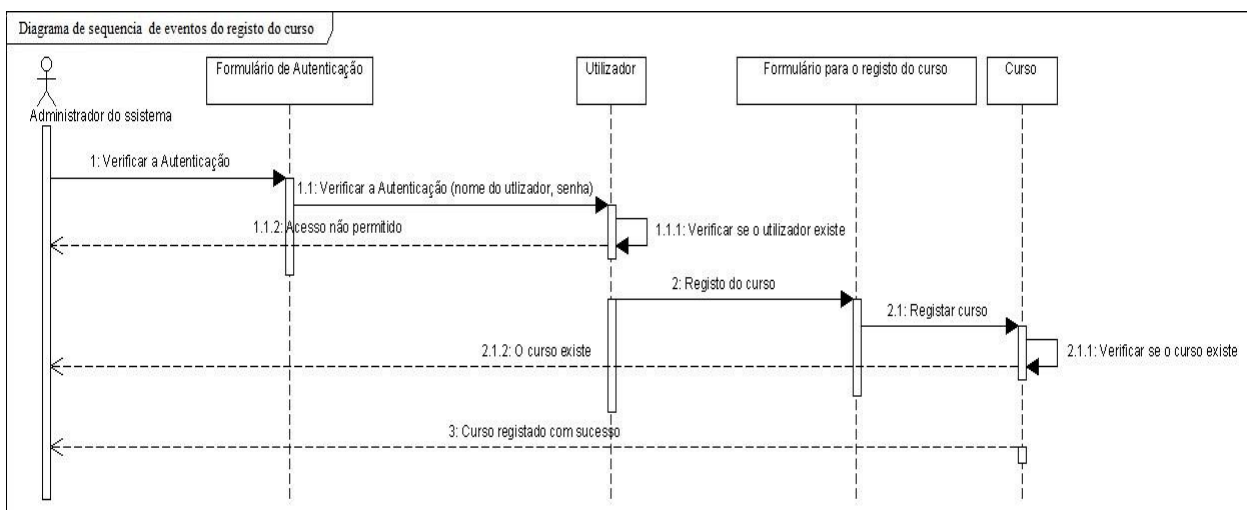


Figura 16 - Diagrama de sequência de eventos para registar o curso

Uma das entidades indispensáveis na programação de horários escolares é o professor, pois esta constitui uma das razões da existência do complexo tema gestão de horários. Na figura 17 apresentamos o diagrama de sequência de eventos desencadeados no seu registo.

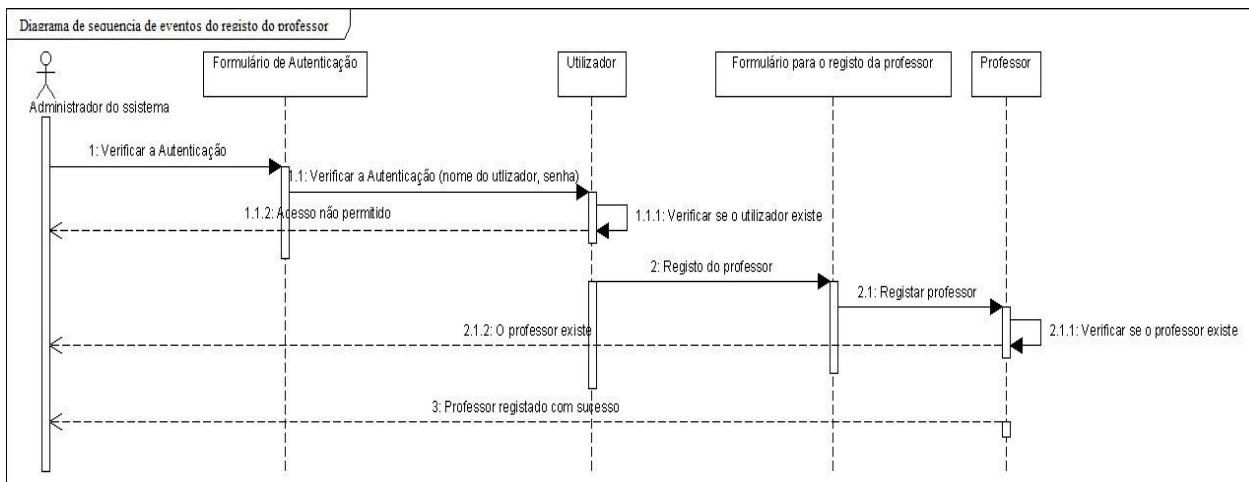


Figura 17 - Diagrama de sequência de eventos para registar o professor

A sala é um recurso físico importante para a realização de aulas nos diferentes horários lectivos.

A figura 18 mostra a sequência de eventos para o seu registo.

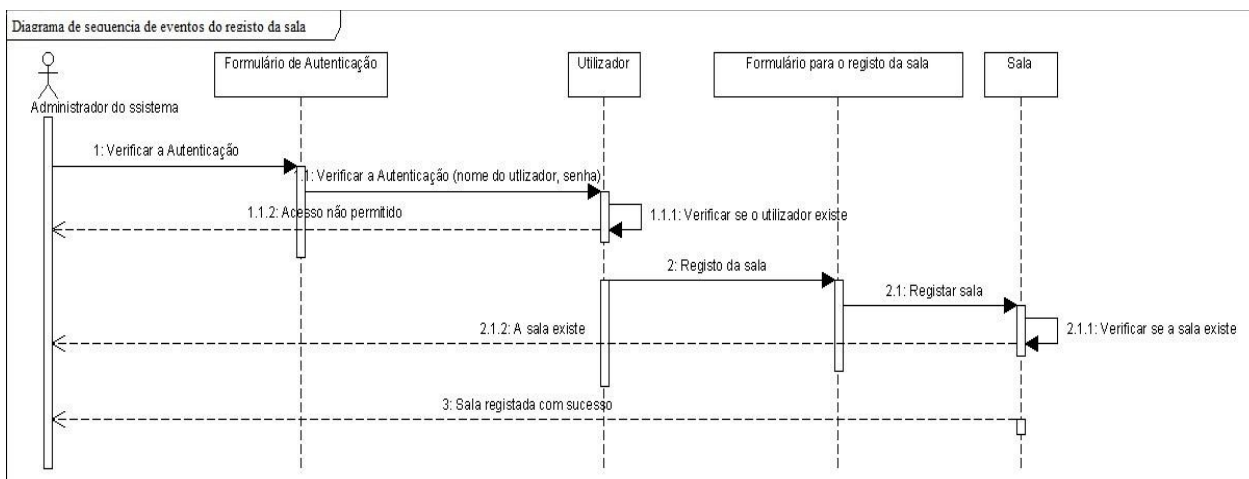


Figura 18 - Diagrama de sequência de eventos para registar a sala

Outra entidade de extrema importância e imprescindível na programação de horários escolares é a turma, pois só existe o horário se existir a turma. A figura 19 ilustra a sequência de eventos para o registo desta entidade.

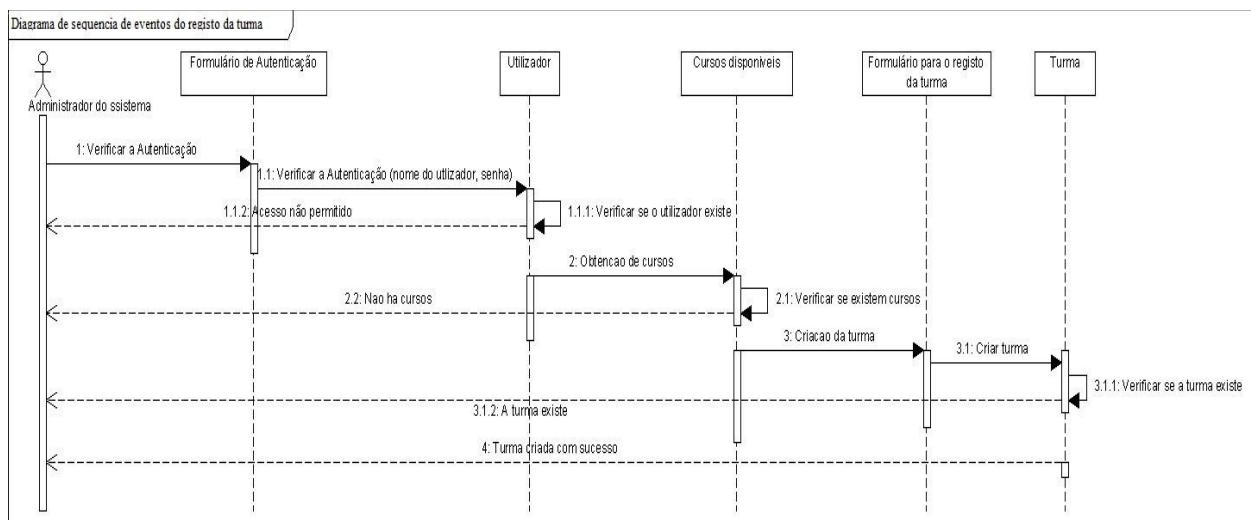


Figura 19 - Diagrama de sequência de eventos para o registo da turma

A seguir passamos a apresentar na figura 20, o diagrama de sequência de eventos para a criação de horários, instrumento regulador do tempo lectivo, imprescindível e objecto de estudo do presente trabalho.

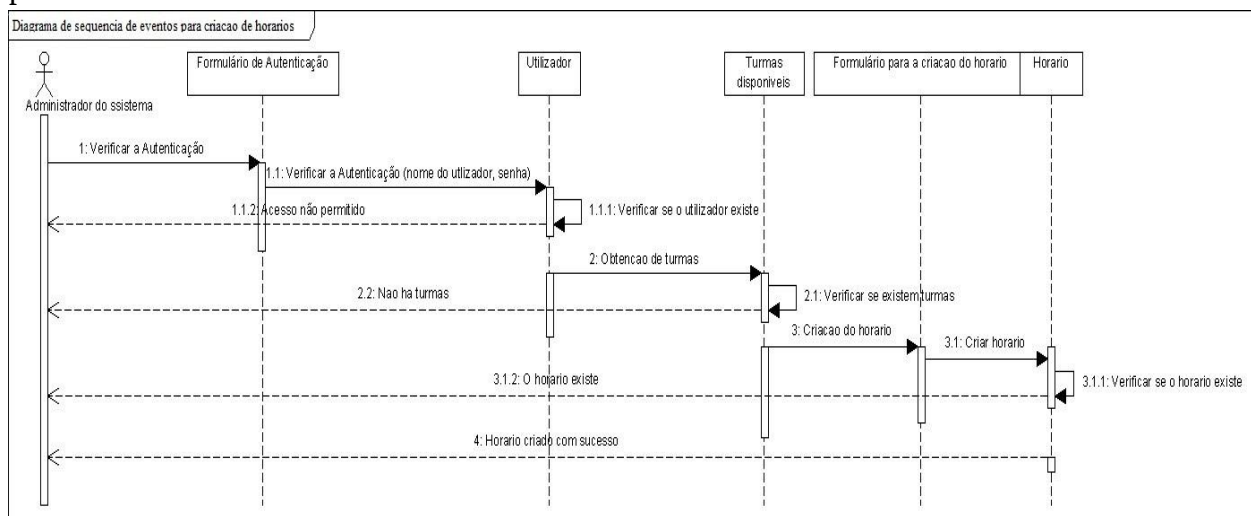


Figura 20 - Diagrama de sequência de eventos para o registo do horário

6. Conclusões e recomendações

Este trabalho apresenta um algoritmo para o problema de programação de horários. O algoritmo proposto, combina a meta-heurística Busca Tabu e o método de busca local iterada, para produzir uma solução óptima do problema de programação de horários.

A Busca Tabu é uma técnica que surgiu de conceitos da inteligência artificial e é um procedimento de busca que explora o espaço de soluções além do óptimo local. Esta busca permite passar para uma solução da vizinhança de cada configuração corrente mesmo que não seja tão boa como a corrente, escapando de óptimos locais e continuando a busca para soluções melhores.

De acordo com os resultados da pesquisa, nota-se que a programação de horários recorrendo-se a métodos exactos, como por exemplo a programação inteira, ainda é ineficaz pois o tempo computacional por eles consumido é demasiadamente elevado. Assim a solução óptima do problema de programação de horários é facilmente obtida por métodos heurísticos, apesar de não garantirem a melhor solução.

Face à necessidade de elaboração de horários pelas instituições educacionais, pode-se afirmar que os objectivos deste trabalho foram na sua maioria alcançados. De modo específico, o software desenvolvido mostra-se oportuno na optimização e prevenção de conflitos nos horários, isto é, a solução óptima é produzida em pouco tempo computacional, sem violar as restrições fortes.

Em todos os métodos, não é respeitada, por completo, a preferência de todos professores no que diz respeito aos horários sem janelas e às aulas geminadas, para poder satisfazer as restrições mais fortes. Este resultado não é surpreendente, pois as heurísticas nem sempre conseguem gerar a solução óptima.

Por se tratar de uma fase inicial, é recomendando que sejam feitos testes exaustivos do sistema para apurar e reportar os eventuais erros ou falhas que o sistema possa apresentar, de maneiras que sejam corrigidos.

Recomenda-se que o software seja executado em um computador com um mínimo de 100 Megabytes de memória RAM livres.

Recomenda-se que os utilizadores do sistema sejam treinados para poderem no usar o software.

Recomenda-se que sejam feitos estudos para que se faça ampliação do sistema para abranger a também a programação de horários de exames.

Referências Bibliográficas

- Alvarez-Valdés, R.; Martin, G. & Tamarit, M. (1996) *Constructing Good Solutions For the Spanish School Timetabling Problem*. *Journal of the operational Research Society*, 47, pp. 1203–1215.
- Avella, P. & Vasil'ev, I. (2005) *A Computational Study Of a Cutting Plane Algorithm for University Course Timetabling*. *Journal of Scheduling*, 8, pp. 497–514.
- Abramson, D.; Abela, J. (1992) *A parallel genetic algorithm for solving the school timetabling problem. em: proceedings of the 15th australian computer science conference*.
- Abramson, D. (1991) *Constructing school timetables using simulated annealing: sequential and parallel algorithms*. *Management science*, 37,1, pp. 98-113.
- Bardadym, V. A (1996) *Computer-Aided School and University Timetabling: The New Wave*, In Burke, E.K. and Ross, P. (eds), *Lecture Notes in ComputerScience*, 1153:22-45, Springer-Verlag.
- Bez, E. T. (2005) *Procedimento de Representação de Soluções em Optimização Global: Aplicação de modelos de interação espacial*. Tese (Doutorado em Engenharia de Produção). Universidade Federal de Santa Catarina.
- Birbas, T.; Daskalaki S. & Housos, E. (1997) *Timetabling for Greek High Schools*. *Journal of the Operational Research Society*, 48, 2, pp.1191-1200, December 1997.
- Caldeira, J. & Agostinho, C.(1997). *School timetabling using genetic search: Practice and Theory of Automated Timetabling*. Toronto.
- Collorni, A., Dorigo, M. & Maniezzo, V. (1998) *Metaheuristics for High School Timetabling:Computational Optimization and Applications*, 9:275-298.
- Costa, D. (1994) *A tabu search algorithm for computing an operational timetable*. *European*

Journal of Operational Research, 76, pp. 98–110.

Cung, V. (2002) *Strategies For The Parallel Implementation Of Metaheuristics. Em: Essays and Surveys in Metaheuristics*. Kluwer Academic Publishers.

Daskalaki, S. (2004) *An integer programming formulation for a case study in university timetabling*. *European Journal of Operational Research*, 153, pp. 117–135.

Drexl, A. & Salewski, F. (1997) *Distribution requirements and compactness constraints in school timetabling*. *European Journal of Operational Research*, 102, 1, pp. 193–214.

Elisondo, R.; Parada, V.; Pradenas, L. (2010) *An evolutionary and constructive approach to a crew scheduling problem in underground passenger transport*. *Journal of Heuristics*, v.16, n.4, p.575-591.

Ferland, J & Lavoue, A. (1992). *Discrete applied mathematics: Exchange procedures for*

timetabling problems.

Glover, F. (1997) *Tabu search and adaptive memory programming advances, applications and challenges*, *Interfaces in Computer Science and Operations Research*. Boston: Kluwer, 1997.

Hertz, A. & Werra, D. (1987) *Computing: Using tabu search techniques for graph coloring*.

Kirkpatrick, S.; Gelatt, C. D. & Vecchi, M. P. (1983) *Optimization by simulated annealing*. *Science*, 220, pp. 671–680.

Lobo, E. L. M. (2005). *Uma solução do problema de horário escolar via algoritmo genético*

Paralelo (tese de mestrado) no Centro Federal Tecnológico de Minas Gerais

Papoutsis, K.; Valouxis, C. & Housos, E. (2003) *A Column Generation Approach For The Timetabling Problem of Greek High Schools*. *Journal of the Operational Research Society*, 54, pp. 230–238.

Schaerf, A. (1999) *Local search techniques for large high school timetabling problems*. *IEEE Transactions on Systems*, 29, 4, pp. 368–377.

Souza, M. J.(2000). *Programação de Horários em Escalas: Uma aproximação por Metaheurísticas*. Thesis, Rio de Janeiro.

Stefano, C. D. & Tettamanzi, A. (2001) *An Evolutionary Algorithm For Solving The School Time-Tabling Problem*. *Proceedings of the Evoworkshops on applications of evolutionary computing*. London, UK: Springer-Verlag.

Torreão, J. R. A.; Roe, E. (1995) *Microcanonical optimization applied to visual processing*.

Physics Letters, 205, pp. 377–382.

UnitsExpress. (2012). *Gerador de Horários (Horários em tempo record)*. Consultado em 08 de Maio de 2012, de <http://www.programahorario.com>

Zukowski, J.(2005). *The Definitive Guide to Java Swing(The Expert's Voice In Java)*. 3ª Edição. USA: Apress.

7. Anexos

7.1 Guião de entrevista

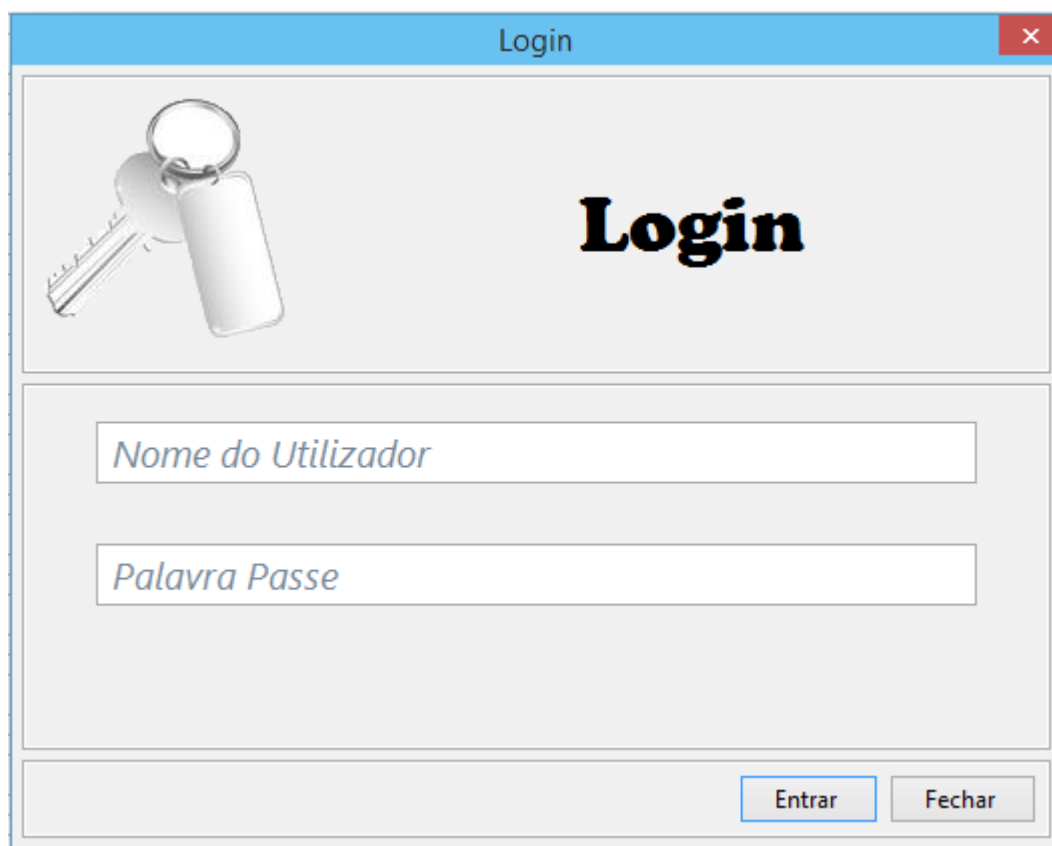
A seguir é apresentado o guião de perguntas feitas na entrevista. Há que realçar que as perguntas foram feitas aos directores adjuntos pedagógicos, com excepção das perguntas 13, 17 e 18 que foram feitas ao universo de 20 professores incluindo os directores adjuntos pedagógicos.

1. Quantos turnos são aplicados na escola?
2. Quais são os horários aplicados nesses turnos?
3. Quanto à estrutura organizacional, como é que a escola está organizada?
4. Que classes tem a escola?
5. Como é que é feita a distribuição dessas classes em termos de turnos?
6. Quantas turmas têm a escola?
7. As turmas têm salas fixas?
8. Quanto tempo dura uma aula?
9. Quanto tempo dura o intervalo?
10. Há intervalo de lanche?
 - a. Quanto tempo dura o intervalo de lanche?
11. A carga horária de uma disciplina é uniforme para todas as secções de uma mesma classe?

12. Qual é o máximo de aulas geminadas permitido por disciplina?
13. Quanto tempo demora a programação de horários?
14. Que requisitos são necessários para a elaboração de horários?
15. Como é que é tratada a questão de indisponibilidade dos professores?
16. Como é que é tratada a questão das preferências dos professores?
17. Como classifica a construção manual da grade horária (difícil, razoável, fácil)?
18. Os horários produzidos satisfazem as necessidades dos professores em termos de aplicabilidade?

7.2 Interface gráfica

Nesta secção apresentamos a interface gráfica do sistema que interagirá com o utilizador, para fazer a gestão de horários escolares. A figura 21 mostra-nos a tela em que se faz a autenticação de utilizadores para poderem ter o acesso ao sistema. Nesta tela o utilizador introduz os seus credenciais carregando posteriormente o botão entrar.



The image shows a standard Windows-style login window titled "Login". The window has a light blue title bar with a red close button in the top right corner. The main content area is light gray and contains an illustration of a key on the left and the word "Login" in a large, bold, black serif font on the right. Below this, there are two white text input fields. The first field is labeled "Nome do Utilizador" in a blue italicized font. The second field is labeled "Palavra Passe" in a blue italicized font. At the bottom right of the window, there are two buttons: "Entrar" (highlighted with a blue border) and "Fechar" (a standard gray button).

Figura 21 – Autenticação de utilizadores

Depois da autenticação do utilizador, é exibida a tela principal do sistema. Nesta tela faz-se a gestão de todas as entidades envolvidas na criação de horários escolares incluindo os próprios horários. A figura 22 mostra a tela principal do sistema.

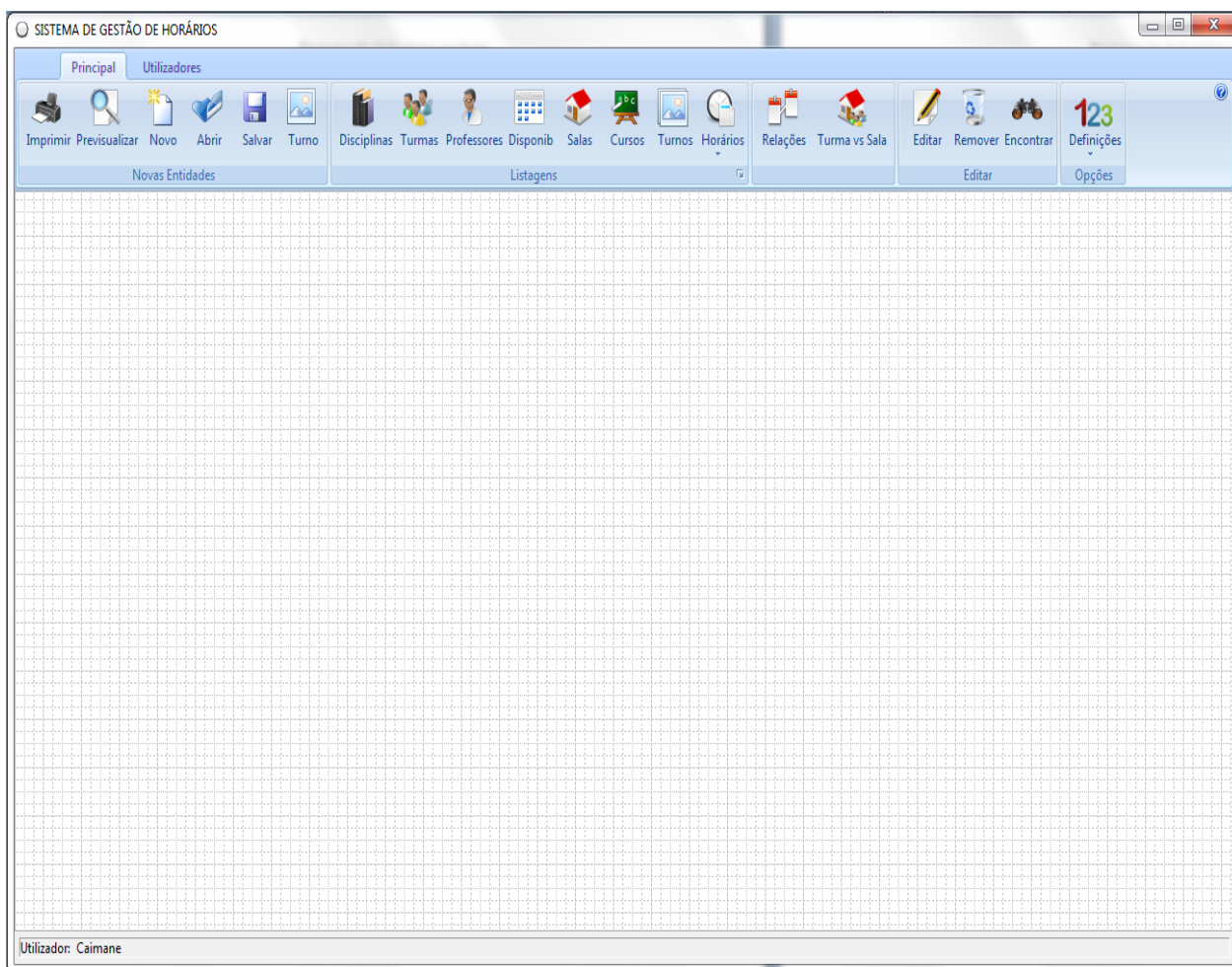



Figura 22 – Tela principal

A seguir é apresentada na figura 23 a tela do registo do professor

Professor

Código
201484736

Registo de Professores



Dados Pessoais

Apelido

Nome

Sexo

Contacto

Telefone

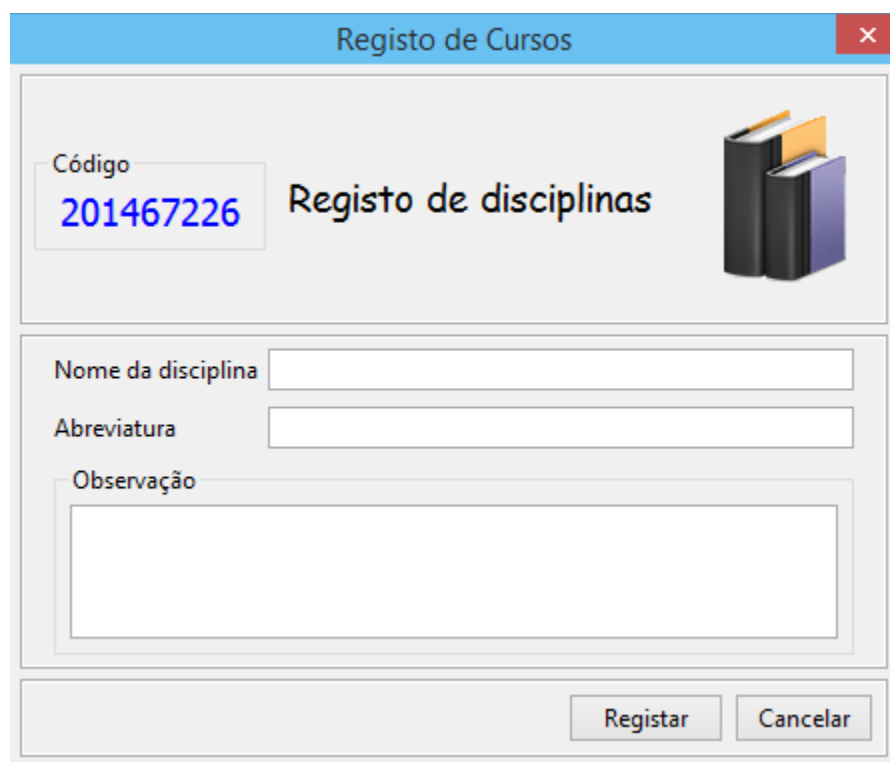
email

Observação

Registrar Cancelar

Figura 23 – Formulário de registo de professores

A figura 24 mostra a tela do registo de disciplinas que posteriormente serão usados na criação de horários.



The image shows a software window titled "Registro de Cursos" with a red close button in the top right corner. Inside the window, there is a light gray panel. On the left of this panel, the word "Código" is above a text box containing the number "201467226". To the right of this, the text "Registro de disciplinas" is displayed. Further right is an icon of two books. Below this panel is a larger form area with three input fields: "Nome da disciplina" (with a text box), "Abreviatura" (with a text box), and "Observação" (with a larger text area). At the bottom right of the window, there are two buttons: "Registrar" and "Cancelar".

Figura 24 – Formulário de registo de disciplinas

A turma é registrada recorrendo-se ao formulário apresentado na figura 25.

Turma

Código
201466246

Registo de Turmas

Nome da Turma

Nr. de alunos:

Curso
Informatica

Nivel

Secção

Turno
Turno da Manhã

Observação

Registrar Cancelar

Figura 25 – Formulário de registo de turmas

Para o registo da sala em serão alocadas as turmas recorre-se ao formulário mostrado na figura 26



Registo de Salas

Código
201490858

Registo de Salas

Nome da Sala

Capacidade:

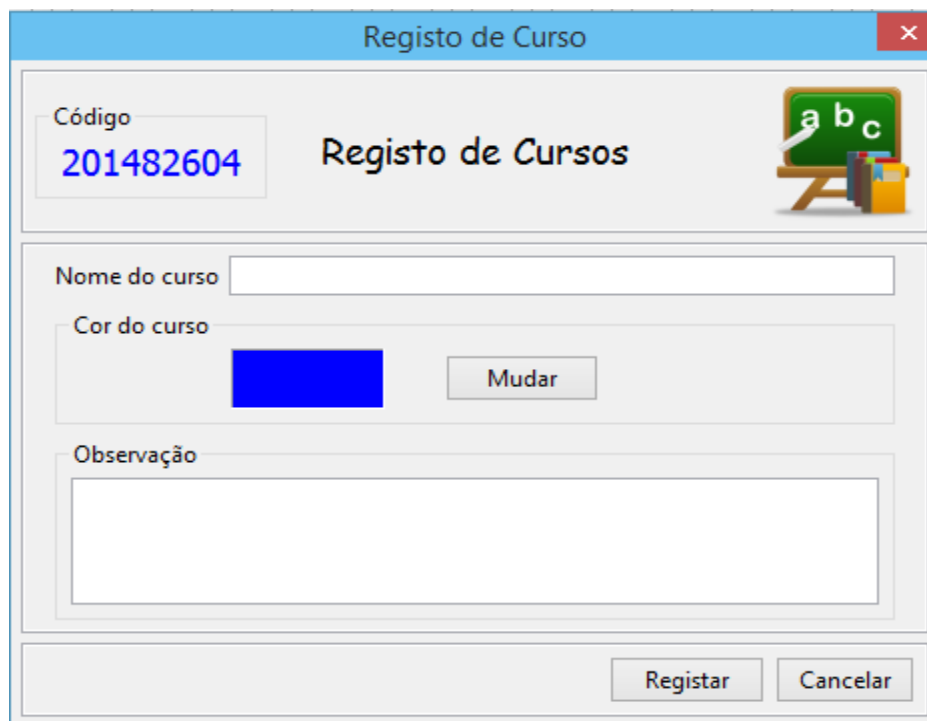
Tipo

Observação

Registrar Cancelar

Figura 26 - Formulário de registo de salas

Na figura 27 é apresentado o formulário do registo do curso



The image shows a software window titled "Registo de Curso". Inside the window, there is a section with a label "Código" and a text box containing the value "201482604". To the right of this is the text "Registo de Cursos" and a small icon of a chalkboard with the letters "a b c". Below this is a section with a label "Nome do curso" and an empty text box. Underneath that is a section with a label "Cor do curso", a blue square color selection, and a "Mudar" button. Below the color section is a section with a label "Observação" and a large empty text area. At the bottom right of the window are two buttons: "Registrar" and "Cancelar".

Figura 27 – Formulário de registo de cursos

Finalmente para a criação do horário, o objectivo do presente trabalho, recorre se ao uso do formulário apresentado na figura 28. Neste são filtradas as turmas através da indicação do turno, o curso (Ensino), nível, e a secção no caso de classes (níveis) com secção. Seguidamente criam-se os horários pisando-se no botão próximo.

Criação de horários
✕

Turnos

Turno da Tarde

Hora de Entrada	Hora de Saida
13h : 25min	14h : 05min
14h : 10min	14h : 50min
14h : 55min	15h : 35min
15h : 40min	16h : 20min
16h : 30min	17h : 10min
17h : 15min	17h : 55min

Intervalo Maior depois do 4º tempo
 Duracao: 10min

Turmas

Curso

Ensino Primario II Ciclo

Nível

Secção

7a Classe

Escolher	Nome da turma	Nível
<input checked="" type="checkbox"/>	Turma 5	7a Classe
<input checked="" type="checkbox"/>	Turma 4	7a Classe
<input checked="" type="checkbox"/>	Turma 3	7a Classe
<input checked="" type="checkbox"/>	Turma 2	7a Classe
<input checked="" type="checkbox"/>	Turma 1	7a Classe
<input checked="" type="checkbox"/>	Turma 6	7a Classe
<input type="checkbox"/>	Turma 7	7a Classe
<input type="checkbox"/>	Turma 8	7a Classe
<input type="checkbox"/>	Turma 9	7a Classe
<input type="checkbox"/>	Turma 10	7a Classe
<input type="checkbox"/>	Turma 11	7a Classe
<input type="checkbox"/>	Turma 12	7a Classe

<
>

<< Anterior

Próximo >>

Cancelar

Figura 28 – Criação de horários

As figuras 29 e 30 mostram respectivamente os exemplos de horários de um professor de Matemática e Ciências Naturais e de uma turma da 7ª

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
1 13h:25 14h:05	Mat (Turma 5)	CN (Turma 5)	CN (Turma 6)	CN (Turma 4)	Mat (Turma 6)
2 14h:10 14h:50	Mat (Turma 5)	CN (Turma 4)	Mat (Turma 5)	CN (Turma 5)	Mat (Turma 6)
3 14h:55 15h:35	Mat (Turma 4)	Mat (Turma 6)	Mat (Turma 5)	CN (Turma 6)	
4 15h:40 16h:20	Mat (Turma 4)	Mat (Turma 6)	Mat (Turma 4)		Mat (Turma 5)
5 16h:30 17h:10	Mat (Turma 6)		Mat (Turma 4)		Mat (Turma 5)
6 17h:15 17h:55	Mat (Turma 6)			Mat (Turma 4)	Mat (Turma 4)

Figura 29 – Exemplo de horário de um dos professores

	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
1 13h:25 14h:05	Mat	CN	MC	SC	Port
2 14h:10 14h:50	Mat	EM	Mat	CN	Port
3 14h:55 15h:35	Port	OF	Mat	EM	OF
4 15h:40 16h:20	Port	EF	I	I	Mat
5 16h:30 17h:10	EV	SC	Port	I	Mat
6 17h:15 17h:55	EV	SC	Port	MC	EF

Figura 30 – Exemplo de horário de uma das turmas