

Universidade Federal de Uberlândia
Faculdade de Engenharia Elétrica
Pós-Graduação em Engenharia Elétrica



Construção e implantação de um sistema
automático de alocação de espaço físico
baseado em computação evolutiva para a
Universidade Federal de Uberlândia

Guilherme Palhares Theodoro

Uberlândia

2016

Universidade Federal de Uberlândia
Faculdade de Engenharia Elétrica
Pós-Graduação em Engenharia Elétrica

Construção e implantação de um sistema
automático de alocação de espaço físico
baseado em computação evolutiva para a
Universidade Federal de Uberlândia

Guilherme Palhares Theodoro

Dissertação apresentada à Universidade Federal
de Uberlândia, perante a banca de examinadores
abaixo, como parte dos requisitos para obtenção
do título de Mestre em Ciências. Avaliada em 18
de Novembro de 2016.

Adélio José de Moraes, Dr.		UFU
Igor Santos Peretta, PhD.		UFU
Keiji Yamanaka, PhD.	Orientador	UFU
Ricardo Soares Bôaventura, Dr.		IFTM

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

T388c
2016 Theodoro, Guilherme Palhares.
 Construção e implantação de um sistema automático de alocação de
 espaço físico baseado em computação evolutiva para a Universidade
 Federal de Uberlândia / Guilherme Palhares Theodoro. - 2016.
 83 f. : il.

 Orientador: Keiji Yamanaka.
 Dissertação (mestrado) - Universidade Federal de Uberlândia,
 Programa de Pós-Graduação em Engenharia Elétrica.
 Inclui bibliografia.

 1. Engenharia elétrica - Teses. 2. Algoritmos genéticos - Teses. 3.
 Administração de instalações - Teses. 4. Universidade Federal de
 Uberlândia - Teses. I. Yamanaka, Keiji. II. Universidade Federal de
 Uberlândia. Programa de Pós-Graduação em Engenharia Elétrica. III.
 Título.

CDU: 621.3

Construção e implantação de um sistema automático de alocação de espaço físico baseado em computação evolutiva para a Universidade Federal de Uberlândia

Guilherme Palhares Theodoro

Dissertação apresentada à Universidade Federal de Uberlândia como parte dos requisitos para obtenção do título de Mestre em Ciências.

Prof. Keiji Yamanaka, PhD.

Orientador

Prof. Darizon Alves de Andrade, PhD.

Coordenador do Curso de Pós-Graduação

Agradecimentos

Ao meu orientador, professor Keiji Yamanaka, pelos ensinamentos, indicação do tema deste trabalho e supervisão durante todas as etapas do mestrado.

Ao professor Igor Santos Peretta, por sua enorme contribuição no desenvolvimento dos artigos relacionados à este trabalho.

À minha família, pelo apoio e compreensão.

Aos meus colegas de trabalho, pela explicação das regras de negócio do problema, além de outras contribuições.

Resumo

Realizado duas vezes por ano letivo, o processo de alocação de salas na Universidade Federal de Uberlândia (UFU) é uma atividade com alto grau de complexidade. A atividade consiste na distribuição de um número elevado de turmas nas salas disponíveis para alocação. O administrador de espaço físico da instituição deve atender uma série de restrições definidas pela direção e, na medida do possível, as necessidades de cada curso. Isso faz com que todo o processo leve meses. O presente trabalho descreve o problema de alocação de salas (PAS) e as particularidades existentes no processo da UFU e de outras universidades, comparando o PAS em diferentes instituições verificou-se que os mesmos lidam com objetivos diferentes e, portanto, não podem ser resolvidos utilizando a mesma abordagem. Um método de busca adequado para problemas de otimização complexos como o PAS, são os algoritmos genéticos (AG). Analisando as maneiras existentes para realizar o tratamento de restrições nos AG juntamente com os objetivos considerados na alocação de salas da UFU, decidiu-se por utilizar uma representação especial do indivíduo além de operadores genéticos específicos para o problema. Um sistema para automatizar o PAS na UFU foi construído e implantado, os resultados obtidos utilizando dados reais da universidade indicaram a viabilidade do método, além da redução do tempo necessário para a realização do processo.

Palavras-chave: Algoritmos Genéticos, Problema de Alocação de Salas

Abstract

Accomplished twice per school year, the rooms allocation process at the Federal University of Uberlândia (UFU) is an activity with a high degree of complexity. The activity consists of the distribution of a large number of courses into available classrooms for allocation. The administrator of institution physical space must attend a number of constraints defined by the direction and, as far as possible, the needs of each course. The constraints make the entire process longer, taking months to be done. The present work describes the space allocation problem (SAP) and the existing particularities in the process of UFU and other universities, it was verified by comparing SAP in different institutions that they handle different objectives and, therefore, they can't be solved using the same approach. A suitable search method for complex optimization problems as SAP, are genetic algorithms (GA). Analyzing the existing ways for handling constraints in GA along with the considered objectives for space allocation in UFU, it was decided to use a special representation of the individual and also specific genetic operators for the problem. A system for automating the SAP in UFU was built and deployed, the obtained results using university's real data indicated the feasibility of the method, besides reducing the time required for carry out the process.

Keywords: Genetic Algorithms, Space Allocation Problem.

Lista de Figuras

1	Modelo matemático da função objetivo UFPB	24
2	Modelo matemático da função objetivo UFSM parte 1	27
3	Modelo matemático da função objetivo UFSM parte 2	28
4	Modelo matemático da função objetivo USP	31
5	Fluxograma dos passos gerais de um AG	39
6	Método de seleção por torneio; em “Torneios”, o indivíduo des- tacado é o ganhador	45
7	Método de seleção por <i>ranking</i>	47
8	Exemplo de pontos de corte	48
9	Cruzamento de 1 ponto	49
10	Cruzamento de 2 pontos	49
11	Cruzamento heurístico uniforme	50
12	Cruzamento heurístico não uniforme	50
13	Mutação em um indivíduo com representação binária	51
14	Curva típica de um AG com elitismo em um problema de mi- nimização	52
15	Representação do indivíduo	56
16	Parte de um indivíduo com dados reais	57
17	Cruzamento heurístico não uniforme para o indivíduo proposto	58
18	Mutação para o indivíduo proposto	59
19	Aplicação 11.02.03.11 - Relação de prédios e cursos	61
20	Aplicação 11.02.03.12 - Relação de turmas e seções aba ‘Ca- dastro’	62

21	Aplicação 11.02.03.12 - Relação de turmas e seções aba ‘Geração Automática’	63
22	Aplicação 11.02.03.12 - Relação de turmas e seções aba ‘Relatório’	63
23	Relatório 11.02.03.12 - Relação turmas seções	64
24	Relatório 11.02.03.12 - Seções críticas	65
25	Relatório 11.02.03.12 - Quantidade de turmas por horário . . .	66
26	Aplicação 11.02.03.13 - Processamento alocação temporária . .	67
27	Aplicação 11.02.03.14 - Alocação temporária	68
28	Aplicação 11.02.03.14 - Choques de alocação destacados na cor vermelha	69
29	Aplicação 11.02.03.14 - Exclusão e realocação de turmas	70
30	Aplicação 11.02.03.14 - Alocação definitiva	71
31	Resultado da execução 2	73

Lista de Tabelas

1	Tamanho das instâncias nas diferentes universidades	32
2	Restrições de qualidade nas diferentes universidades	34
3	Método e propósitos das funções objetivo nas diferentes uni- versidades	35
4	Tratamento de restrições para o problema da UFU	55
5	Parâmetros de execução do AG	60
6	Resultados de execução bloco 5O-A	73
7	Resultados de execução para o 2º semestre de 2016	75

Lista de Abreviaturas e Siglas

AG	Algoritmos Genéticos
AGc	Algoritmo Genético Compacto
CA	Central de Aulas
CT	Centro de Tecnologia
ICMC	Instituto de Ciências Matemáticas e de Computação
PAS	Problema de Alocação de Salas
SG	Sistema de Gestão
UFPB	Universidade Federal da Paraíba
UFSM	Universidade Federal de Santa Maria
UFU	Universidade Federal de Uberlândia
USP	Universidade de São Paulo

Sumário

1	Introdução	13
1.1	Objetivo Geral	15
1.2	Objetivos Específicos	16
1.3	Metodologia	16
1.4	Estrutura da Dissertação	17
2	O Problema de Alocação de Salas	19
2.1	O Processo de Alocação de Salas da UFU	20
2.2	Trabalhos Relacionados	22
2.2.1	Problema de Alocação de Aulas: O Caso da Central de Aulas da UFPB	23
2.2.2	Solução do Problema de Alocação de Salas Utilizando um Modelo Matemático Multi-Índice	25
2.2.3	Aplicação da Metaheurística AGc para o Problema de Alocação de Aulas à Salas	29
2.3	Considerações Finais	32
3	Algoritmos Genéticos	37
3.1	Representação do Indivíduo	40
3.2	Inicialização da população	41
3.3	Função de Avaliação	42
3.4	Seleção	43
3.4.1	Seleção por Roleta	43
3.4.2	Seleção por Torneio	44

3.4.3	Seleção por <i>Ranking</i>	45
3.5	Cruzamento (Recombinação)	47
3.5.1	Cruzamento de 1 Ponto	48
3.5.2	Cruzamento de 2 Pontos	49
3.5.3	Cruzamento Heurístico Uniforme	49
3.5.4	Cruzamento Heurístico Não Uniforme	50
3.6	Mutação	51
3.7	Escolha de Pais e Filhos Para Próxima Geração	51
3.8	Crítérios de parada	52
4	Materiais e Métodos	54
4.1	Funcionamento do Algoritmo Genético	54
4.2	Tecnologias Empregadas	60
4.3	Aplicações Desenvolvidas como Resultado deste Trabalho	60
5	Resultados e Discussões	72
6	Relato do Administrador de Espaços da UFU	77
7	Conclusões	79

Capítulo 1

Introdução

A alocação de espaço físico é um problema de otimização que surge na maioria das instituições acadêmicas. O problema consiste em distribuir pessoas (professores e alunos) e recursos (equipamentos, aulas, etc.) em espaços físicos atendendo uma série de restrições e preferências. O desafio é gerar soluções ótimas em tempo aceitável do ponto de vista prático. A alocação não eficiente de espaços e recursos pode resultar em custos operacionais, transtornos logísticos, insatisfação, entre outros problemas. Devido à complexidade de resolução do problema, a resolução de forma manual pode ser ineficiente em função do não atendimento de todas as restrições e preferências. Além disso, uma alocação de salas de baixa qualidade pode causar insatisfações na comunidade acadêmica [1].

O Problema de Alocação de Salas, segundo Schaefer em [2] é a distribuição de aulas para as devidas salas com horários previamente estabelecidos, respeitando uma série de restrições, tais como infraestrutura adequada das salas e acessibilidade para os alunos e docentes. As restrições e as necessidades de recursos dificultam a distribuição de salas, logo, a solução manual desse problema acarreta em um processo demorado que requer vários dias e pode gerar transtornos para o início das aulas por falta ou erro de alocação, além do fluxo acentuado de alunos em deslocamento.

Para Bardadyn em [3] e Carter e Tovey em [4], o PAS faz parte de um conjunto de problemas de programação de horários em cursos universitários (*course timetabling*). A conferência PATAT (*International Conference on the Practice and Theory of Automated Timetabling*) que realizou sua 11^a edição no ano de 2016, trata especificamente de problemas de automatização envolvendo cursos universitários e reforça a relevância

do tema proposto nesse trabalho para a comunidade acadêmica.

Na Universidade Federal de Uberlândia, o processo de alocação de espaço físico ocorre duas vezes por ano. Até o primeiro semestre letivo do ano de 2016, o processo era feito manualmente por um administrador de espaços com a ajuda de até dois estagiários consumindo um total de 640 horas de trabalho.

A automatização desse processo de alocação permitiria que o administrador de espaços da instituição economizasse tempo e esforços. Além disso, em um cenário onde várias soluções pudessem ser obtidas com pouco tempo computacional, o administrador ganharia tempo para decidir sobre a alocação mais adequada, considerando as restrições e preferências da instituição.

Schaefer em [2] e de Werra em [5] entendem que essa classe de problemas não pode ser completamente automatizada. Existem duas justificativas para isto: por um lado, há razões, que não podem ser facilmente expressas em um sistema automatizado que tornariam uma alocação melhor que a outra. Por outro, uma vez que o espaço de soluções é vasto, a intervenção humana pode conduzir a busca em direção a regiões promissoras, nas quais o sistema, por si só, dificilmente teria condições de chegar.

Burke e Varley em [6] propõem a utilização de três técnicas metaheurísticas para a automatização do PAS: subida da encosta (*hill-climbing*), recozimento simulado (*simulated annealing*) e algoritmos genéticos. Foi demonstrado que a utilização dessas técnicas é uma forma promissora de tratar o problema de alocação de espaços em universidades e que, quanto maior o número de restrições, menor é a probabilidade de se obter uma solução aceitável.

Uma tendência recente tem sido o desenvolvimento de heurísticas híbridas que combinam certas características de uma heurística com outra, com o objetivo de melhorar o desempenho pela superação das fraquezas de uma heurística ou de ambas [7].

Apesar dos grandes avanços no projeto de algoritmos de otimização sofisticados, ainda existe uma grande preocupação com relação à sua utilidade na prática. De fato, o desenvolvimento dos computadores permitiu com que os pesquisadores projetassem metaheurísticas eficientes para resolver problemas difíceis de otimização combinatória. Entretanto, muitos desses algoritmos focam na resolução de problemas acadêmicos encontrados na literatura que muitas vezes não capturam a complexidade de problemas reais [8].

O trabalho de McCollum em [8] indica claramente que esse autor ficou surpreso em descobrir que até então existiam pouquíssimos artigos sobre o tema relatando que os métodos pesquisados foram implementados e utilizados pela instituição.

A comparação do presente trabalho com outros como Burke e Varley em [6], Cirino et al. em [9], Henan e Shao-Wen em [10], Landa Silva em [11], Queiroga et al. em [12] e Sales et al. em [13] é uma tarefa com resultados inconclusivos, uma vez que as restrições tratadas e as instâncias de problema em cada um desses trabalhos são diferentes. Para Paechter em [14], mesmo quando existem critérios claros para julgar diferentes algoritmos de automação para uma mesma instância de problema, ainda podem existir dificuldades de se interpretar os resultados significativamente. Paechter também enfatiza sobre a necessidade de se trabalhar com a pessoa que utiliza o sistema automatizado para melhorar o entendimento de se uma solução pode ser considerada boa ou não.

1.1 Objetivo Geral

O objetivo geral deste trabalho é construir e implantar um sistema baseado em computação evolutiva para automatizar o processo de alocação de salas da Universidade Federal de Uberlândia. O sistema construído deve atender todas as restrições que são

atendidas pelo processo de alocação de espaço físico feito manualmente.

1.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Realizar pesquisa bibliográfica sobre o problema de alocação de espaço físico e sobre algoritmos genéticos;
- Descrever as etapas que envolvem o processo de alocação manual realizado na UFU e levantar os requisitos necessários para construção do sistema;
- Projetar um sistema que para automatização do processo de alocação de salas da UFU que utilize algoritmos genéticos;
- Construir o sistema e apresentar suas interfaces;
- Verificar a viabilidade de aplicação do método com o sistema construído utilizando dados reais;
- Implantar o sistema e apresentar os resultados provenientes de sua utilização para a alocação de salas no 2º semestre de 2016;

1.3 Metodologia

O motivo principal que determina a realização da pesquisa nesse trabalho é de ordem prática, pois decorre da necessidade de tornar mais eficiente o processo de alocação de salas da UFU.

A pesquisa realizada no presente trabalho enquadra-se no campo das pesquisas Teórico-Aplicada Descritiva. O principal objetivo da pesquisa descritiva é a descrição

das características de determinada população ou fenômeno ou, então, o estabelecimento de relações entre variáveis, cuja principal característica reside na utilização de técnicas padronizadas para a coleta de dados, tais como o questionário e a observação sistemática.

A adoção deste conceito parece adequada para a realização desta pesquisa, dado que o objetivo é desenvolver um sistema para automatizar o processo de alocação de salas da UFU, que proporcionaria a execução desse processo de maneira mais rápida e eficiente.

Em um primeiro momento, este trabalho consiste de uma pesquisa bibliográfica sobre o assunto em questão, com a finalidade de buscar informações sobre a utilização atual, bem como ter uma visão geral dos trabalhos já realizados. Em seguida será realizado um estudo de campo, pois o processo de alocação de salas da UFU será investigado a fundo proporcionando que o pesquisador compreenda todas as necessidades do processo atual.

Serão utilizados dados reais referente à oferta de disciplinas da UFU no primeiro e segundo semestre de 2016.

1.4 Estrutura da Dissertação

O trabalho está dividido em 7 capítulos da seguinte forma:

- **Capítulo 1 - Introdução:** apresentação das considerações iniciais, objetivos, justificativa e metodologia;
- **Capítulo 2 - O Problema de Alocação de Salas:** contextualização do problema de alocação de salas, exposição do caso da UFU e de outras universidades;
- **Capítulo 3 - Algoritmos Genéticos:** revisão dos conceitos de Algoritmos

Genéticos e suas características fundamentais;

- **Capítulo 4 - Materiais e métodos:** interpretação computacional para o problema da UFU utilizando algoritmos genéticos e a apresentação das aplicações criadas;
- **Capítulo 5 - Resultados e Discussões:** apresentação da análise dos resultados para o conjunto de dados utilizado;
- **Capítulo 6 - Relato do Administrador de Espaços da UFU:** exibição da contribuição do sistema proposto para o processo de alocação de salas da universidade;
- **Capítulo 7- Conclusões:** apresentação da conclusão sobre a pesquisa realizada.

Capítulo 2

O Problema de Alocação de Salas

O problema de alocação de salas em uma instituição acadêmica é definido como a atribuição de um conjunto finito de aulas, distribuídas em diferentes intervalos de tempo, em um conjunto também finito de salas, respeitando uma série de restrições operacionais e de preferências [4]. O problema considera que existem horários preestabelecidos do início e término das aulas e um conjunto de salas onde estas ocorrerão. Nesse sentido, o problema incide em determinar uma alocação das turmas em salas de aula de forma que requisitos considerados essenciais sejam acatados, tornando viável uma ou mais soluções[15].

Para Carter e Tovey em [4], o PAS é um problema de otimização combinatória. Problemas de otimização combinatória são caracterizados por um número finito de soluções admissíveis e são comuns na vida cotidiana e na engenharia. Uma área importante de aplicações está relacionada à utilização eficiente de recursos escassos para aumentar a produtividade. Alguns problemas de otimização combinatória encontrados na engenharia são: empacotamento, mochila, agendamento de máquinas, roteamento de veículos, caixeiro viajante, entre outros [16].

Em grande parte dos casos estudados, o problema tem sido classificado como NP-difícil, o que na prática pode inviabilizar sua resolução por processo manual ou por algoritmos determinísticos para grande volume de dados, pois o espaço de soluções cresce exponencialmente inviabilizando assim uma solução ótima em tempo razoável. Nesses casos, uma alternativa viável é a resolução por técnicas heurísticas ou algoritmos aproximativos que, apesar de não garantirem soluções ótimas, podem gerar boas soluções com um custo computacional razoável e adequado às necessidades da aplicação

[3].

2.1 O Processo de Alocação de Salas da UFU

A Universidade Federal de Uberlândia possui campi nas cidades de Uberlândia, Ituiutaba, Patos de Minas e Monte Carmelo todos no estado de Minas Gerais. Apenas na cidade sede, são 245 salas de aula distribuídas em 3 campi e 14 blocos (prédios) diferentes.

O processo de alocação de salas da UFU acontece uma vez por semestre letivo, sendo que em cada um mais de 4000 turmas devem ser distribuídas no espaço físico disponível. Feito manualmente, o processo leva aproximadamente 640 horas para ser executado.

O processo de alocação é iniciado assim que a oferta de disciplinas para o próximo semestre é finalizada. Na oferta de disciplinas, cada coordenação de curso indica as disciplinas que serão disponibilizadas para matrícula no próximo semestre. Cada turma ofertada exige o preenchimento das seguintes informações:

- Curso ofertante;
- Código e nome da disciplina;
- Período ideal (período no qual a disciplina se encontra no currículo do curso);
- Código da turma;
- Quantidade de vagas oferecidas;
- Dias da semana;
- Horário de realização;

- Data de início e fim do período;
- Necessidade de alocação de sala em espaço físico comum da universidade (SIM ou NÃO);
- Necessidade de utilização de pranchetas (SIM ou NÃO).

Apenas as salas em comum da universidade são disponibilizadas para alocação. Algumas unidades acadêmicas possuem suas próprias salas e laboratórios que podem ser utilizados para as disciplinas ofertadas. Nesses casos, deve ser indicado que não existe necessidade de alocação de sala para a turma.

A maior parte das coordenações de curso monta sua grade horária tentando privilegiar os alunos que cursam apenas as disciplinas de determinado período. Dessa forma, os horários das disciplinas de um mesmo período são encaixados para que esses alunos tenham aulas sempre em um mesmo turno (matutino, vespertino ou noturno) e não haja horários vagos. Mas nem todas as coordenações conseguem planejar os horários dessa forma.

Vale ressaltar ainda que, por inúmeros fatores (reprovações, adiantamento de disciplinas, entre outros), casos de alunos que não cursam apenas as disciplinas de seu período ideal são frequentes.

O principal objetivo considerado no processo de alocação de salas é fazer a alocação de espaço físico por período de cada curso, ou seja, acomodar todas as disciplinas de um determinado período de um curso em uma mesma sala, desde que os horários se encaixem. Um objetivo secundário é tentar alocar todas as turmas de um curso em um único bloco. Esses objetivos tentam minimizar o deslocamento entre salas de alunos e professores.

Levando em consideração as necessidades de cada curso, as características das salas disponíveis em cada bloco e a proximidade entre os blocos, o administrador de espaços

da instituição criou uma relação de cursos e blocos. Essa relação indica para qual bloco as turmas de um determinado curso e período devem ser direcionadas.

A partir da lista de disciplinas ofertadas e da relação de cursos e blocos, o administrador de espaços cria um plano de alocação de modo que as disciplinas de um mesmo período e curso sejam alocadas em uma mesma sala sempre que possível.

Os objetivos considerados no processo manual de alocação de salas são transformados nos seguintes requisitos para construção do sistema:

- Utilizar a relação de cursos e blocos para a alocação de espaço físico;
- É necessário alocar todas as turmas ofertadas que indicarem a necessidade de alocação de sala em espaço físico comum;
- Alocar turmas do mesmo curso e período em uma única sala quando os horários estiverem encaixados;
- Em uma mesma sala, horário e período não pode haver mais de uma turma;
- Uma sala não pode receber uma turma cuja quantidade de alunos seja superior à sua capacidade;
- Turmas sem necessidade de prancheta devem ser alocadas apenas em salas com carteiras; turmas com necessidade de prancheta devem ser alocadas apenas em salas com pranchetas.

2.2 Trabalhos Relacionados

Esta seção do capítulo apresenta três trabalhos relacionados ao tema dessa dissertação que foram publicados na 47^a edição do Simpósio Brasileiro de Pesquisa Operacional

(XLVII SBPO) no ano de 2015. O simpósio é um evento anual promovido pela Sociedade Brasileira de Pesquisa Operacional.

Foram selecionados apenas trabalhos de universidades brasileiras por possuírem maior semelhança com a cultura educacional da Universidade Federal de Uberlândia.

Os trabalhos descritos a seguir fornecem evidências da importância, complexidade e diversidade do problema de alocação de salas.

2.2.1 Problema de Alocação de Aulas: O Caso da Central de Aulas da UFPB

Os autores Queiroga, Júnior e Cabral em [12] propõem a utilização de um modelo baseado em Programação Linear Inteira para resolver o PAS na Central de Aulas (CA) da Universidade Federal da Paraíba (UFPB). A tarefa era realizada manualmente e as soluções obtidas eram repletas de inconsistências, tais como choques de horário e desconsideração das capacidades das salas.

Na UFPB, em boa parte dos casos, a alocação das aulas é feita em duas etapas: em um primeiro momento, os gestores de cada centro tentam alocar suas demandas de aulas utilizando as instalações (salas de aula/laboratórios) que lhes são disponíveis. Caso a quantidade de aulas com necessidade de alocação exceda o número de salas disponíveis em cada centro, a demanda excedente é encaminhada para ser alocada na central de aulas. Assim, no geral, as salas da CA recebem as aulas excedentes que não puderam ser alocadas nas instalações dos seus respectivos centros de origem [12].

O estudo de caso foi realizado utilizando dados do segundo semestre letivo de 2014, sendo que 1131 aulas de 645 turmas deveriam ser alocadas em 60 salas de 7 blocos diferentes [12].

As principais restrições no problema foram [12]:

- Todas as aulas devem ser alocadas em uma sala em um determinado horário;
- Duas aulas não podem ser alocadas em uma mesma sala no mesmo horário/dia da semana;
- A quantidade de alunos de uma turma não pode exceder a capacidade da sala;
- O indicador de qualidade considerado para análise da solução foi o número de aulas de uma mesma turma alocadas em salas diferentes.

A função objetivo do trabalho visa minimizar o número de aulas de uma mesma turma alocadas em salas diferentes [12]. O modelo matemático proposto é apresentado na Figura 1.

<i>Conjuntos</i>	
T	Conjunto das turmas t a serem alocadas
H_t	Conjunto das aulas (horários) i de cada turma t
S	Conjunto das salas j disponíveis
Ch	Conjunto dos choques de horário c entre as aulas
H_c	Conjunto das aulas (horários) k do choque de horário c
<i>Dados de Entrada</i>	
Q_j	Capacidade da sala j
D_t	Demanda da turma t
<i>Variáveis de Decisão</i>	
$x_{ij} = \begin{cases} 1, & \text{se a aula } i \text{ for alocada na sala } j \\ 0, & \text{caso contrário} \end{cases}$	
$y_t = \begin{cases} 1, & \text{se as aulas da turma } t \text{ não forem alocadas na mesma sala} \\ 0, & \text{caso contrário} \end{cases}$	
Modelo	
	$\min \sum_{t \in T} y_t \quad (1)$
s.a	$\sum_{j \in S} x_{ij} = 1 \quad \forall i \in H_t t \in T \quad (2)$
	$\sum_{k \in H_c} x_{kj} \leq 1 \quad \forall j \in S, \forall c \in Ch \quad (3)$
	$x_{ij} D_t \leq Q_j \quad \forall j \in S, \forall i \in H_t t \in T \quad (4)$
	$y_t \geq x_{ij} - x_{kj} \quad \forall t \in T \forall i, k \in H_t, \forall j \in S \quad (5)$
	$x_{ij} \in \{0, 1\} \quad \forall i \in P, \forall j \in T_i \quad (6)$
	$y_t \in \{0, 1\} \quad \forall t \in T \quad (7)$

Figura 1: Modelo matemático da função objetivo UFPB [12]

Os autores concluem que o modelo matemático proposto se mostrou eficiente na resolução do PAS. O resultado da alocação manual foi comparado com o resultado da alocação automática e a solução produzida automaticamente apresentou poucas inconsistências pois: proibiu a existência de choques de horário, reduziu o estouro das capacidades existentes, fez uma melhor utilização da capacidade das salas além de alocar em uma mesma sala aulas de uma mesma turma [12].

2.2.2 Solução do Problema de Alocação de Salas Utilizando um Modelo Matemático Multi-Índice

Os autores Sales, Müller e Simonetto em [13] propõem desenvolver um modelo matemático adequado para o PAS do Centro de Tecnologia (CT) da Universidade Federal de Santa Maria (UFSM) .

O CT-UFSM hospeda 12 cursos de graduação. O campus possui cinco blocos, contendo um total de 50 salas de aula: 3 laboratórios, 2 salas com mesas de desenho alta, 1 sala com mesas de desenho baixa e as demais com mesas escolares. As salas possuem capacidade entre 25 e 50 alunos e possuem quadros do tipo branco e verde [13].

As restrições do problema são divididas em restrições essenciais e restrições de qualidade [13].

As restrições essenciais são:

- Toda disciplina deve ser alocada em cada período que ocupa;
- A sala pode ter no máximo 1 disciplina em cada horário;
- A capacidade da sala deve ser maior/igual que a oferta de vagas da disciplina;
- As disciplinas que requerem mesas de desenho alta devem ficar nesse tipo de sala;

- As disciplinas que requerem mesas de desenho baixa devem ficar nesse tipo de sala.

As restrições de qualidade são:

- As salas designadas devem estar próximas da unidade acadêmica de origem;
- Manter o professor na mesma sala de aula na maior parte do tempo;
- Professores que desejam salas com quadro branco não devem ficar em salas com quadro verde;
- Professores que desejam salas com quadro verde não devem ficar em salas com quadro branco.

A função objetivo do trabalho busca minimizar a quantidade de assentos vazios, além de diminuir a distância percorrida pelos professores [13]. O modelo matemático proposto é apresentado nas Figuras 2 e 3.

m – representa a quantidade de disciplinas a serem alocadas.
 n – representa a quantidade de salas disponíveis para alocação.
 l – representa a quantidade de períodos (*slots*) de tempo disponíveis para alocação.
 D – representa o conjunto das disciplinas i a serem alocadas.
 S – representa o conjunto das salas j disponíveis para alocação.
 P – representa o conjunto de períodos j disponíveis para alocação.
 C_j – representa a capacidade, em termos de número de alunos, da sala j .
 R_j – representa o tipo de equipamento que está instalado na sala j (1 com quadro verde, 2 com quadro branco, 3 com mesas de desenho baixas, 4 com mesas de desenho altas, 6 com quadro verde e branco).
 P_i – representa o número de vagas solicitadas para a disciplina i .
 IS_i – representa o período de início da disciplina i (por exemplo se ele inicia na segunda-feira pela manhã às 9h 30 min, então $IS_i = 3$).
 NS_i – representa o número de períodos da disciplina i (ela pode ser de 1, 2, 3, 4 ou 5 horas).
 SR_i – representa as necessidades especiais da disciplina i , por exemplo se ela necessita de mesa de desenho alta, $SR_i = 4$).
 d_{ij} – representa a distância do departamento ao qual a disciplina i pertence e a sala j .
 OS_i – representa o conjunto dos períodos de tempo que a disciplina i ocupa, por exemplo se ele inicia na segunda-feira pela manhã às 9h 30 min e ocupa dois períodos, então $OS_i = \{3,4\}$.
 VI – representa o conjunto dos pares (i, j) , ou seja (disciplina i , sala j), para os quais a restrição de capacidade é respeitada, ou seja, $(C_j - P_i \geq 0)$.
 DDA – representa o conjunto das disciplinas i que necessitam de mesa de desenho alta, ou seja, $SR_i = 4$.
 DDB – representa o conjunto das disciplinas i que necessitam de mesa de desenho baixa, ou seja, $SR_i = 3$.
 SDA – representa o conjunto das salas j que possuem de mesa de desenho alta, ou seja, $R_j = 4$.
 SDB – representa o conjunto das salas j que possuem de mesa de desenho baixa, ou seja, $R_j = 3$.
 DP_k – representa o conjunto de disciplinas i que ocupam o período k , ou seja, $(IS_i + NS_i - 1) \geq k$ e $IS_i \leq k$.
 $D1H$ – representa o conjunto de disciplinas i que tem carga horária de 1 hora, por período de distribuição, ou seja, $NS_i = 1$.
 $D2H$ – representa o conjunto de disciplinas i que tem carga horária de 2 horas, por período de distribuição, ou seja, $NS_i = 2$.
 $D3H$ – representa o conjunto de disciplinas i que tem carga horária de 3 horas, por período de distribuição, ou seja, $NS_i = 3$.
 $D4H$ – representa o conjunto de disciplinas i que tem carga horária de 4 horas, por período de distribuição, ou seja, $NS_i = 4$.
 $D5H$ – representa o conjunto de disciplinas i que tem carga horária de 5 horas, por período de distribuição, ou seja, $NS_i = 5$.
 $x_{ijk} = \begin{cases} 1, & \text{se a disciplina } i \text{ está alocada a sala } j \text{ no período } k \\ 0, & \text{caso contrário} \end{cases}$, representando a variável binária de decisão do problema.

Figura 2: Modelo matemático da função objetivo UFSM parte 1 [13]

$$\text{Min} \sum_{(i,j) \in VI} \sum_{k \in OS_i} ((C_j/P_i) + d_{ij}) * x_{ijk} \quad (1)$$

Sujeito a

$$\sum_{j=1}^n x_{ijk} = 1, \forall i \in DP_K \text{ e } \forall k = 1, \dots, l \quad (2)$$

$$\sum_{j \in \{S \setminus SDA\}} \sum_{k \in OS_i} x_{ijk} = 1, \forall i \in \{D1H \setminus DDA\} \quad (3)$$

$$\sum_{j \in \{S \setminus SDA\}} \sum_{k \in OS_i} x_{ijk} = 2, \forall i \in \{D2H \setminus DDA\} \quad (4)$$

$$\sum_{j \in \{S \setminus SDA\}} \sum_{k \in OS_i} x_{ijk} = 3, \forall i \in \{D3H \setminus DDA\} \quad (5)$$

$$\sum_{j \in \{S \setminus SDA\}} \sum_{k \in OS_i} x_{ijk} = 4, \forall i \in \{D4H \setminus DDA\} \quad (6)$$

$$\sum_{j \in \{S \setminus SDA\}} \sum_{k \in OS_i} x_{ijk} = 5, \forall i \in \{D5H \setminus DDA\} \quad (7)$$

$$\sum_{j \in SDA} \sum_{k \in OS_i} x_{ijk} = 1, \forall i \in \{DIH \cap DDA\} \quad (8)$$

$$\sum_{j \in SDA} \sum_{k \in OS_i} x_{ijk} = 2, \forall i \in \{D2H \cap DDA\} \quad (9)$$

$$\sum_{j \in SDA} \sum_{k \in OS_i} x_{ijk} = 3, \forall i \in \{D3H \cap DDA\} \quad (10)$$

$$\sum_{j \in SDA} \sum_{k \in OS_i} x_{ijk} = 4, \forall i \in \{D4H \cap DDA\} \quad (11)$$

$$\sum_{j \in SDA} \sum_{k \in OS_i} x_{ijk} = 5, \forall i \in \{D5H \cap DDA\} \quad (12)$$

$$\sum_{j \in SDB} \sum_{k \in OS_i} x_{ijk} = 1, \forall i \in \{D1H \cap DDB\} \quad (13)$$

$$\sum_{j \in SDB} \sum_{k \in OS_i} x_{ijk} = 2, \forall i \in \{D2H \cap DDB\} \quad (14)$$

$$\sum_{j \in SDB} \sum_{k \in OS_i} x_{ijk} = 3, \forall i \in \{D3H \cap DDB\} \quad (15)$$

$$\sum_{j \in SDB} \sum_{k \in OS_i} x_{ijk} = 4, \forall i \in \{D4H \cap DDB\} \quad (16)$$

$$\sum_{j \in SDB} \sum_{k \in OS_i} x_{ijk} = 5, \forall i \in \{D5H \cap DDB\} \quad (17)$$

$$(C_j - P_i) * x_{ijk} \geq 0, \forall i = 1, \dots, m, \forall j = 1, \dots, n \text{ e } \forall k = 1, \dots, l \quad (18)$$

$$\sum_{i=1}^m x_{ijk} \leq 1, \forall j = 1, \dots, n \text{ e } \forall k = 1, \dots, l \quad (19)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+1)}, \forall i \in D2H \text{ e } \forall j = 1, \dots, n \quad (20)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+1)}, \forall i \in D3H \text{ e } \forall j = 1, \dots, n \quad (21)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+2)}, \forall i \in D3H \text{ e } \forall j = 1, \dots, n \quad (22)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+1)}, \forall i \in D4H \text{ e } \forall j = 1, \dots, n \quad (23)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+2)}, \forall i \in D4H \text{ e } \forall j = 1, \dots, n \quad (24)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+3)}, \forall i \in D4H \text{ e } \forall j = 1, \dots, n \quad (25)$$

$$x_{ij(IS_i+1)} = x_{ij(IS_i+2)}, \forall i \in D4H \text{ e } \forall j = 1, \dots, n \quad (26)$$

$$x_{ij(IS_i+1)} = x_{ij(IS_i+3)}, \forall i \in D4H \text{ e } \forall j = 1, \dots, n \quad (27)$$

$$x_{ij(IS_i+2)} = x_{ij(IS_i+3)}, \forall i \in D4H \text{ e } \forall j = 1, \dots, n \quad (28)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+1)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (29)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+2)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (30)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+3)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (31)$$

$$x_{ij(IS_i)} = x_{ij(IS_i+4)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (32)$$

$$x_{ij(IS_i+1)} = x_{ij(IS_i+2)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (33)$$

$$x_{ij(IS_i+1)} = x_{ij(IS_i+3)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (34)$$

$$x_{ij(IS_i+1)} = x_{ij(IS_i+4)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (35)$$

$$x_{ij(IS_i+2)} = x_{ij(IS_i+3)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (36)$$

$$x_{ij(IS_i+2)} = x_{ij(IS_i+4)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (37)$$

$$x_{ij(IS_i+3)} = x_{ij(IS_i+4)}, \forall i \in D5H \text{ e } \forall j = 1, \dots, n \quad (38)$$

$$x_{ijk} \in \{0, 1\}, \forall i = 1, \dots, m, \forall j = 1, \dots, n \text{ e } \forall k = 1, \dots, l \quad (39)$$

Figura 3: Modelo matemático da função objetivo UFSM parte 2 [13]

Os resultados computacionais, obtidos a partir do *solver* IBM ILOG CPLEX Optimization Studio 12.4 indicaram uma melhor distribuição dos espaços de modo que em muitas das soluções houve sobra de salas em comparação com a solução manual vigente [13].

2.2.3 Aplicação da Metaheurística AGc para o Problema de Alocação de Aulas à Salas

Os autores Cirino, Santos e Delbem em [9] propõem a resolução do PAS do Instituto de Ciências Matemáticas e de Computação (ICMC) da Universidade de São Paulo (USP) por meio das heurísticas: Algoritmo Genético Compacto (AGc) e Algoritmo de Estimulação de Distribuição.

O ICMC-USP possui 23 salas de aula distribuídas em 4 blocos, onde são ministradas cerca de 260 aulas para 140 turmas. No estudo, cada turma está associada a um ou mais perfis de alunos que são conjuntos de discentes que dividem o mesmo curso e ano de ingresso e, portanto, estão cursando um conjunto preestabelecido de turmas [9].

As restrições do problema estão divididas em restrições essenciais e restrições de qualidade [9].

As restrições essenciais são:

- Todas as aulas devem ser alocadas em alguma sala;
- As aulas não podem ser sobrepostas na sala, ou seja, duas aulas com sobreposição de horário não podem ser simultaneamente alocadas na mesma sala;
- As aulas só poderão ser alocadas em salas que atendam suas demandas de requisitos e capacidade para acomodar todos os participantes.

As restrições de qualidade são:

- Alocar aulas em salas que possuem capacidade o mais próximo possível do número de inscritos da aula;
- Alocar todas as aulas de uma mesma turma em uma mesma sala;
- Reduzir o deslocamento dos perfis de aluno, ou seja, alocar todas as aulas dos perfis em salas próximas umas das outras;
- Evitar o uso de salas preferencialmente vazias;
- Alocar as aulas dos perfis respeitando as preferências por salas dos mesmos.

A função objetivo do trabalho busca minimizar: o percentual de assentos vazios nas salas de aula, o número de troca de salas de uma mesma turma, a preferência dos perfis, o uso de salas preferencialmente vazias e o deslocamento das turmas considerando as distâncias entre as salas [9]. O modelo matemático proposto é apresentado na Figura 4.

Conjuntos considerados:

H	Conjunto de <i>slots</i> de Horário ($h \in \{1, \dots, H\}$);
R	Conjunto de recursos ($r \in \{1, \dots, R\}$);
P	Conjunto dos perfis de turma ($p \in \{1, \dots, P\}$);
S	Conjunto de salas ($s \in \{1, \dots, S\}$);
T	Conjunto de turmas ($t \in \{1, \dots, T\}$).

Dados de entrada:

ρ_{th}	Matriz binária de turma t por horário h ;
σ_{sr}	Matriz binária de salas s e recursos r ;
τ_{tr}	Matriz binária de turmas t e recursos r ;
ν_{tp}	Matriz binária de turma t por perfis p ;
$\varphi_{ss'}$	Matriz de distâncias entre salas s e s' ;
χ_{ps}	Matriz de preferências dos perfis p por salas s ;
CAP_s	Capacidade máxima da sala s ;
PV_s	Vetor binário indicando se sala s é preferencialmente vazia;
TAM_t	Quantidade de alunos inscritos na turma t .

Dados de preprocessamento:

A	Conjunto de aulas ($a \in \{1, \dots, A\}$);
t_a	Turma $t \in T$ da qual a aula a faz parte;
$\theta_t \subseteq A$	Conjunto de aulas da turma t ;
$\mu_p \subseteq A$	Conjunto de aulas do perfil p ;
$CH^a \subseteq A$	Conjunto de aulas que possuem conflito com aula a ;
$S^* \subseteq S$	Conjunto de salas preferencialmente vazias;
η_{as}	Matriz binária de aulas a por salas s , aonde $\eta_{as} = 1$ indica que aula a pode ser alocada na sala s ;
c_{as}	Matriz de custo de alocação, $c_{as} \doteq [100 \cdot (1 - TAM_{t_a}/CAP_s)]$.

Variáveis de decisão:

x_{as}	Variável binária indicando se aula a esta alocada na sala s ;
y_t	Variável inteira indicando o número de trocas de sala da turma t ;
w_{ps}	Variável binária indicando se perfil p possui aulas alocadas na sala s ;
k_{psi}	Variável binária indicando se perfil p possui aulas alocadas nas salas s, i .

Modelo Proposto:

$$\min \quad z = \alpha \left(\sum_{a \in A} \sum_{s \in S} c_{as} x_{as} \right) + \beta \left(\sum_{t \in T} y_t \right) + \gamma \left(\sum_{p \in P} \sum_{s=1}^{|S|-1} \sum_{t=s+1}^{|S|} \varphi_{st} k_{psi} \right) + \delta \left(\sum_{a \in A} \sum_{s \in S^*} x_{as} \right) + \varepsilon \left(\sum_{p \in P} \sum_{s \in S} \chi_{ps} w_{ps} \right) \quad (I)$$

$$\text{s.a.} \quad \sum_{s \in S} x_{as} = 1 \quad a \in A \quad (II)$$

$$x_{as} \leq \eta_{as} \quad a \in A, s \in S \quad (III)$$

$$x_{as} + x_{a's} \leq 1 \quad a \in A, s \in S, a' \in CH^a \quad (IV)$$

$$\sum_{n=1}^{|\theta_t|} x_{a_n s_n} \leq 1 + y_t \quad t \in T, a_n \in \theta_t, \{s_1, \dots, s_n\} \subseteq S \quad (V)$$

$$x_{as} \leq w_{ps} \quad a \in A, s \in S, p \in \mu_p \quad (VI)$$

$$2k_{psi} \leq w_{ps} + w_{pi} \quad p \in P, s \in S, i \in S \quad (VII)$$

$$k_{psi} \geq w_{ps} + w_{pi} - 1 \quad p \in P, s \in S, i \in S \quad (VIII)$$

$$x_{as} \in \{0,1\} \quad a \in A, s \in S \quad (IX)$$

$$y_t \in \mathbb{Z}_+ \quad t \in T \quad (X)$$

$$w_{ps} \in \{0,1\} \quad p \in P, s \in S \quad (XI)$$

$$k_{psi} \in \{0,1\} \quad p \in P, s \in S, i \in S \quad (XII)$$

Figura 4: Modelo matemático da função objetivo USP [9]

Os resultados foram obtidos com a utilização de 3 instâncias fictícias e uma instância real. A análise final foca na comparação entre as diferentes heurísticas [9].

2.3 Considerações Finais

Como já mencionado no Capítulo 1, a comparação deste trabalho com outros é uma tarefa com resultados inconclusivos, uma vez que as restrições tratadas e as instâncias de dados utilizadas são diferentes das restrições e instâncias da UFU. O restante desse capítulo tem a finalidade de mostrar que os trabalhos lidam com instâncias de tamanhos diferentes, além de enaltecer as características únicas desse trabalho que o diferencia dos outros.

A Tabela 1 apresenta o tamanho das instâncias de cada um dos trabalhos relatados.

Tabela 1: Tamanho das instâncias nas diferentes universidades

	UFU	CA-UFPB	CT-UFSM	ICMC-USP
Turmas(UFU)/Aulas(Outros)	4000	1131	Não indicado	260
Salas	245	60	50	23
Blocos	14	7	5	4

Um fator a ser destacado é que, diferentemente deste, em nenhum dos trabalhos relacionados é mencionado que o sistema de automatização proposto foi realmente adotado pelas respectivas universidades.

Com relação às restrições, algumas delas são comuns em todos os trabalhos que abordam o tema e são responsáveis pelas características essenciais do PAS. As restrições comuns são listadas abaixo:

- Todas as turmas devem ser alocadas em alguma sala;
- Em uma mesma sala, horário e período não pode haver mais de uma turma;

- Uma sala não pode receber uma turma cuja quantidade de alunos seja superior à sua capacidade.

Algumas das restrições de qualidade mencionadas nos trabalhos relacionados podem ser equiparadas com a relação de cursos e blocos da UFU, uma vez que esta foi criada levando em conta as características das salas disponíveis em cada bloco e a proximidade entre eles. A Tabela 2 indica essas restrições:

Tabela 2: Restrições de qualidade nas diferentes universidades

UFU	CA-UFPB	CT-UFSM	ICMC-USP
Adequação à relação de cursos e blocos	Nenhuma	As salas designadas devem estar próximas do departamento de origem	Alocar aulas em salas que possuem capacidade o mais próximo possível do número de inscritos da aula
		Professores que desejam salas com quadro branco não devem ficar em salas com quadro verde	Reduzir o deslocamento dos perfis de aluno, ou seja, alocar todas as aulas dos perfis em salas próximas umas das outras
		Professores que desejam salas com quadro verde não devem ficar em salas com quadro branco	Alocar as aulas dos perfis respeitando as preferências por salas dos mesmos

Apesar dos trabalhos relacionados utilizarem heurísticas diferentes, todos eles possuem uma função objetivo. Essas funções apresentadas nas Figuras 1, 2, 3 e 4 expressam cada uma das restrições tratadas nos trabalhos e isso faz com que elas sejam demasiadamente complexas. Um outro ponto diferencial desse trabalho é que a função objetivo proposta lida exclusivamente com a quantidade de choques de horário. A Tabela 3 faz um resumo dos métodos de busca e propósitos das funções objetivo nos diferentes trabalhos.

Tabela 3: Método e propósitos das funções objetivo nas diferentes universidades

Instituição	Método	Função objetivo
UFU	Algoritmos genéticos	Minimizar a quantidade de choques de alocação
CA-UFPB	Programação linear inteira	Minimizar o número de aulas de uma mesma turma alocadas em salas diferentes
CT-UFSM	Modelo matemático multi-índice	Minimizar a quantidade de assentos vazios, além de diminuir a distância percorrida pelos docentes
ICMC-USP	Algoritmo genético compacto e algoritmo de estimulação de distribuição	Minimizar o percentual de assentos vazios nas salas de aula, o número de troca de salas de uma mesma turma, a preferência dos perfis, o uso de salas preferencialmente vazias e o deslocamento das turmas considerando as distâncias entre as salas

As outras restrições são tratadas pela representação do indivíduo, as operações genéticas específicas e a utilização de expressões condicionais. No Capítulo 4 encontram-se explicações detalhadas sobre o tratamento de restrições desse trabalho.

Capítulo 3

Algoritmos Genéticos

Algoritmos genéticos são algoritmos de busca baseados nos mecanismos de seleção natural e genética. Eles combinam a sobrevivência do mais apto com uma forma estruturada de troca de informação genética entre dois indivíduos para formar uma estrutura heurística de busca [17].

Os AGs foram desenvolvidos por John Holland e seus colegas da Universidade de Michigan. Os objetivos de suas pesquisas foram dois: (1) abstrair e explicar rigorosamente o processo adaptativo de sistemas naturais e (2) projetar sistemas artificiais que mantêm os mecanismos importantes dos sistemas naturais. Esta abordagem trouxe descobertas importantes na ciência de sistemas artificiais e naturais [18].

O tema central da pesquisa em algoritmos genéticos tem sido a robustez, o equilíbrio necessário entre eficiência e eficácia para a sobrevivência em diferentes ambientes. As implicações da robustez para sistemas artificiais são diversas. Se sistemas artificiais podem ser feitos mais robustos, muitas reformulações dispendiosas podem ser reduzidas ou eliminadas. Se altos níveis de adaptação podem ser alcançados, os sistemas já existentes podem realizar suas funções de melhor forma [18].

AGs pertencem à classe de algoritmos probabilísticos, pois eles combinam elementos de busca direta e estocástica. Por causa disso, o AG é mais robusto que outros métodos de busca direta. Enquanto o AG mantém a população de soluções potenciais, todos os outros métodos processam um único ponto no espaço de busca. Por todos esses atributos, os AGs são destinados para problemas complexos de otimização como escalonamento, problemas de transporte, problemas do caixeiro viajante, otimização de consultas de banco de dados, entre outros [19].

AGs pertencem a família de modelos computacionais inspirados na evolução que incorporam uma solução potencial para um problema específico, em uma estrutura semelhante a de um cromossomo e aplicam operadores de seleção e cruzamento de forma a preservar informações críticas relativas à solução do problema. Deve ser observado que cada cromossomo, chamado de indivíduo no AG, corresponde a um ponto no espaço de soluções do problema de otimização [17].

De acordo com Michalewicz em [19], um algoritmo genético para um determinado problema deve conter os seguintes componentes:

- Uma representação genética para as potenciais soluções do problema;
- Uma maneira de criar uma população inicial de possíveis soluções;
- Uma função de avaliação que se comporta como o ambiente, classificando soluções em termos de suas aptidões (*fitness*);
- Operadores genéticos (recombinação e mutação) que alteram a composição dos filhos (soluções descendentes);
- Valores para vários parâmetros que os algoritmos genéticos utilizam (tamanho da população, probabilidade de aplicação de operadores genéticos, entre outros).

A principal ideia dos algoritmos genéticos é criar uma população de indivíduos e, durante certo número de iterações (gerações), evoluir esta população através de auto-adaptação e recombinação [11]. Os passos gerais de um AG são:

1. Criar uma população inicial aleatoriamente ou através de heurística apropriada;
2. Avaliar os indivíduos da população atual a partir de uma função de avaliação;
3. Selecionar indivíduos para serem pais, segundo sua aptidão;

4. Realizar cruzamento entre os indivíduos selecionados para a geração de uma nova população;
5. Realizar mutação dos novos indivíduos;
6. Escolher indivíduos entre pais e filhos para formar a população da nova geração;
7. Se o critério de parada for satisfeito, a execução é finalizada; caso contrário, retorna-se ao passo 2.

O fluxograma da Figura 5 ilustra os passos gerais de um AG.

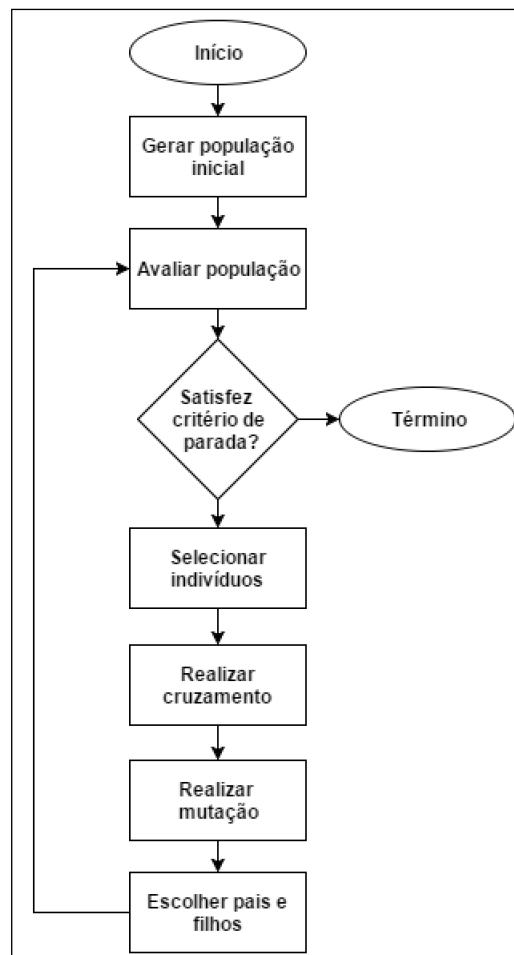


Figura 5: Fluxograma dos passos gerais de um AG

3.1 Representação do Indivíduo

Em algoritmos genéticos, um indivíduo é uma solução, uma solução parcial ou um conjunto de soluções. A representação do indivíduo em algoritmos genéticos é chamada de cromossomo. A escolha de um cromossomo apropriado é um assunto importante, porque tal representação deve ser adequada para o funcionamento eficaz dos operadores genéticos e também pode ajudar a lidar com as restrições do problema. Representações comuns para cromossomos utilizam *strings* binárias. Em muitos casos, estruturas mais complexas são projetadas para representar indivíduos para problemas do mundo real [18].

Para muitos problemas na engenharia é quase impossível representar soluções pela representação binária. Por causa disso, vários métodos de codificação foram criados em uma variedade de problemas para fornecer uma implementação efetiva dos algoritmos genéticos. De acordo com o tipo de símbolo que os alelos de um gene podem assumir, os métodos de codificação podem ser classificados como [16]:

- Codificação binária;
- Codificação por números reais;
- Codificação por números inteiros ou permutação literal;
- Codificação pela estrutura de dados geral.

A codificação por números reais é a mais adequada para problemas de otimização de funções. Foi confirmado por inúmeros desenvolvedores que a codificação por números reais tem melhor desempenho que a codificação binária para a otimização de funções [16].

A codificação por números inteiros ou permutação literal é melhor utilizada para problemas de otimização combinatória, uma vez que a essência dos problemas de otimização combinatória é a busca pela melhor combinação de itens submetidos à determinadas restrições [16].

Para problemas reais mais complexos é sugerido uma estrutura de dados apropriada como o alelo do gene, garantindo a captura da natureza do problema [16].

3.2 Inicialização da população

Os AGs são algoritmos de busca estocásticos baseados em população. Cada AG mantém uma população de soluções candidatas. O primeiro passo na aplicação do AG em um problema de otimização é gerar uma população inicial. O método mais comum na geração de uma população inicial é atribuir um valor aleatório do domínio permitido para cada gene no cromossomo. O objetivo da atribuição de números aleatórios é garantir que a população inicial seja uma representação uniforme de todo o espaço de busca. Se regiões do espaço de busca não forem cobertas pela inicialização da população, então existem chances de que essas regiões sejam negligenciadas pelo processo de busca [20].

O tamanho da população inicial influencia na complexidade computacional e nas habilidades de exploração. Um grande número de indivíduos aumenta a diversidade, melhorando as habilidades de exploração da população. Entretanto, quanto mais indivíduos, maior é a complexidade computacional por geração. Ao passo que o tempo de execução por geração aumenta, pode ser o caso em que uma menor quantidade de gerações sejam necessárias para localizar uma solução aceitável. Por outro lado, uma população pequena representará uma pequena parte do espaço de busca. Nesse caso, o tempo de execução por geração é pequeno, mas o AG pode precisar de mais gerações

para convergir [20].

3.3 Função de Avaliação

No modelo de evolução Darwiniano, os indivíduos com as melhores características possuem melhores chances de sobreviver e de se reproduzir. Com o intuito de determinar a habilidade de um indivíduo sobreviver, uma função matemática é utilizada para avaliar a solução representada por um cromossomo. A função de avaliação (*fitness function*), f mapeia a representação do cromossomo em um valor escalar [20].

A função de avaliação é a maneira utilizada pelos AGs para determinar a qualidade de um indivíduo como solução do problema em questão. A função de avaliação deve, portanto, ser escolhida com cuidado. Ela deve incluir todo o conhecimento que se possui sobre o problema a ser resolvido, tanto com relação às suas restrições quanto aos seus objetivos de qualidade [17].

Para Engel em [20], a função de avaliação representa a função objetivo que, por sua vez, descreve um problema de otimização. Os diferentes tipos de problemas de otimização influenciam na formulação da função de avaliação. Dessa forma:

- **Problemas sem restrições:** a função de avaliação é simplesmente a função objetivo.
- **Problemas com restrições:** para resolver este tipo de problema, alguns AGs mudam sua função de avaliação para conter dois objetivos, o objetivo original e a penalidade referente à restrição.
- **Problemas multi-objetivo:** podem ser resolvidos usando uma abordagem de agregação ponderada, onde a função de avaliação é uma soma ponderada de todos os sub-objetivos, ou pela utilização de algoritmos de otimização baseados

nas fronteiras de Pareto.

3.4 Seleção

O princípio por trás dos algoritmos genéticos é essencialmente a seleção natural de Darwin. A seleção fornece a força condutora (conhecida como pressão evolucionária ou pressão seletiva) em um algoritmo genético. Com muita pressão, a busca genética irá terminar prematuramente (convergência prematura); com pouca pressão, o progresso evolucionário será mais lento do que o necessário (difícil convergência). A seleção direciona a busca genética para regiões promissoras no espaço de busca [16].

Para formar uma nova população, os indivíduos são selecionados com probabilidade proporcional ao seu *fitness*. Isso garante que o número esperado de vezes que um indivíduo é escolhido é aproximadamente proporcional ao seu desempenho relativo na população, proporcionando aos melhores indivíduos mais chances de se reproduzir [21].

Muitos métodos de seleção já foram propostos, sendo alguns dos métodos mais comuns:

- Seleção por Roleta;
- Seleção por Torneio;
- Seleção por *Ranking*.

3.4.1 Seleção por Roleta

A roleta é um tipo de seleção proporcional que influencia o processo de seleção em direção aos indivíduos mais aptos. Após a avaliação f_i de cada indivíduo em uma determinada geração e assumindo que n é o tamanho da população, o *fitness* total da

população é obtido através da Equação (1):

$$S = \sum_{i=1}^n f_i \quad (1)$$

Em seguida, uma probabilidade é designada para cada indivíduo da população, conforme a Equação (2):

$$p_i = \frac{f_i}{S} \quad (2)$$

Finalmente, uma probabilidade acumulada é obtida para cada indivíduo pela adição de sua probabilidade com a probabilidade acumulada dos membros anteriores, conforme a Equação (3):

$$c_i = \sum_{k=1}^i p_k, i = 1, 2, \dots, n \quad (3)$$

Um número aleatório r distribuído uniformemente no intervalo $[0, 1]$ é sorteado por n vezes e, em cada vez, um indivíduo i é selecionado tal que $c_{i-1} < r < c_i$. Quando $r < c_1$ o primeiro indivíduo é selecionado. Esse processo pode ser visualizado como o giro de uma roleta dividida em n pedaços, em que cada pedaço possui tamanho proporcional ao *fitness* do indivíduo.

3.4.2 Seleção por Torneio

O método de torneio consiste em selecionar uma série de indivíduos da população e fazer com que eles entrem em competição pelo direito de ser pai, usando como arma a sua avaliação [17].

Nesse método existe um parâmetro denominado, tamanho do torneio (k), que define quantos indivíduos são selecionados aleatoriamente dentro da população para competir entre si. Uma vez definidos os competidores, aquele dentre eles, que possuir a melhor avaliação é selecionado para aplicação do operador genético [17].

O valor mínimo de k é igual a 2, pois do contrário não haverá competição. Se for escolhido o valor igual ao tamanho da população (n) o vencedor será sempre o

mesmo (o melhor indivíduo) e se forem escolhidos valores altos (próximo ao tamanho da população), os $n-k$ indivíduos tenderão a predominar, uma vez que sempre um deles será vencedor [17].

A Figura 6 exemplifica o método de seleção por torneio.

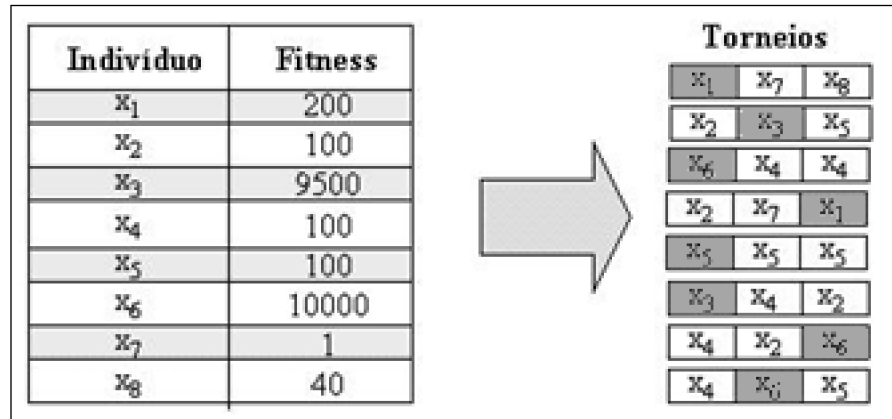


Figura 6: Método de seleção por torneio; em “Torneios”, o indivíduo destacado é o ganhador [17]

Na Figura 6, o tamanho do torneio k é igual a 3; à esquerda, estão os indivíduos e seus respectivos valores de avaliação; à direita, os oito torneios que aconteceram, com seus ganhadores destacados.

3.4.3 Seleção por *Ranking*

A seleção por *ranking* é um método de seleção que evita a convergência prematura e a dominância de um superindivíduo. O seu princípio consiste em ordenar todos os elementos de acordo com sua função de avaliação e usar este *ranking* como base da seleção, ao invés de usar diretamente o valor da avaliação [17].

Ao utilizar o *ranking* como base para seleção em vez da função de avaliação, este método evita o domínio de um superindivíduo, além de manter a pressão seletiva no mesmo nível, desde a primeira até a última geração, não importando o grau de con-

vergência genética que tenha ocorrido na população no decorrer do algoritmo genético [17].

Após a ordenação de indivíduos, deve-se fazer o seu mapeamento para uma função de avaliação. Uma opção de mapeamento possível é o método linear, em que cada indivíduo recebe uma avaliação igual a [17]:

$$E(i, t) = Min + (Max - Min) \cdot \frac{rank(i, t) - 1}{N - 1} \quad (4)$$

onde:

- $E(i, t)$ é o valor do mapeamento para o indivíduo i na geração t ;
- Min é o valor da avaliação que será atribuído ao indivíduo pior colocado no *ranking*;
- Max é o valor da avaliação que será atribuído ao indivíduo melhor colocado no *ranking*;
- N é o número de indivíduos na população;
- $rank(i, t)$ é o *ranking* do indivíduo i na população mantida pelo AG na geração t ;

Uma vez definidos os novos valores de avaliação destes indivíduos, um método tradicional, tal como o da roleta viciada, pode ser utilizado para escolha dos pais que serão submetidos aos operadores genéticos. A Figura 7 ilustra a seleção por *ranking* com $Min=0,9$ e $Max=1,1$.

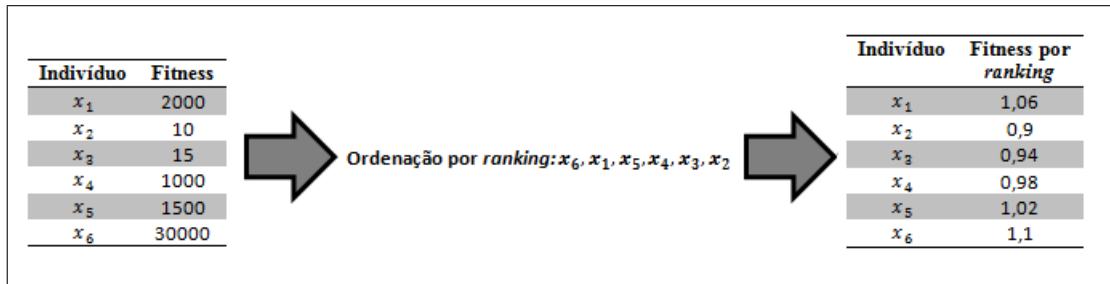


Figura 7: Método de seleção por *ranking* [17]

O problema deste e de todos os métodos que reduzem a pressão seletiva é que o AG pode levar um tempo maior para convergir para uma solução com uma alta função de avaliação. Entretanto, a manutenção da diversidade na população garante que o AG varra um pedaço maior do espaço de soluções, ficando menos suscetível à captura de máximos locais [17].

3.5 Cruzamento (Recombinação)

O cruzamento é o operador de recombinação mais importante, ele utiliza dois indivíduos, chamados de pais, e produz dois novos indivíduos, chamados de filhos, através da troca de partes dos cromossomos dos pais. Através do cruzamento, a busca é direcionada para regiões promissoras do espaço de busca [21], ou seja, é o operador determinante da convergência do algoritmo.

Landa Silva em [11] testou quatro formas de cruzamento para o PAS:

- Cruzamento de 1 ponto;
- Cruzamento de 2 pontos;
- Cruzamento heurístico uniforme;
- Cruzamento heurístico não uniforme;

3.5.1 Cruzamento de 1 Ponto

Depois de selecionados dois pais pelo método de seleção, um ponto de corte é sorteado. Um ponto de corte constitui uma posição entre dois genes de um cromossomo. Cada indivíduo de n genes contém $n-1$ pontos de corte, e este ponto de corte é o ponto de separação entre cada um dos genes que compõem o material genético de cada pai [17]. A Figura 8 mostra os 4 possíveis pontos de corte de um cromossomo com 5 genes.

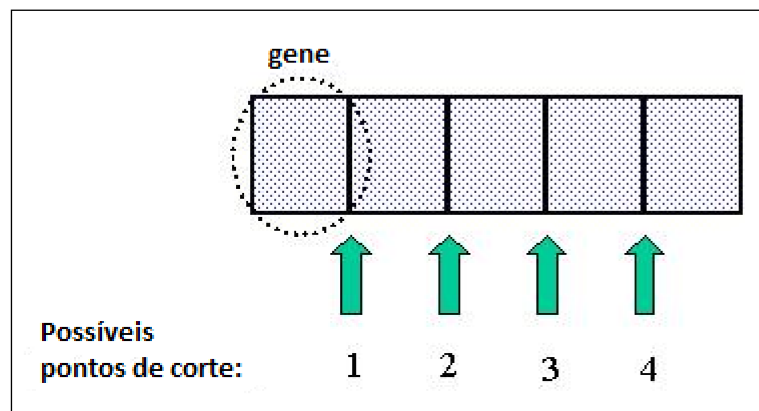


Figura 8: Exemplo de pontos de corte [17]

Depois de sorteado o ponto de corte, os pais são separados em duas partes: à esquerda e à direita do ponto de corte. O primeiro filho é composto através da concatenação da parte do primeiro pai à esquerda do ponto de corte com a parte do segundo pai à direita do ponto de corte. O segundo filho é composto através da concatenação das partes que sobraram (a metade do segundo pai à esquerda do ponto de corte com a metade do primeiro pai à direita do ponto de corte) [17]. A figura 9 exemplifica essa forma de cruzamento.

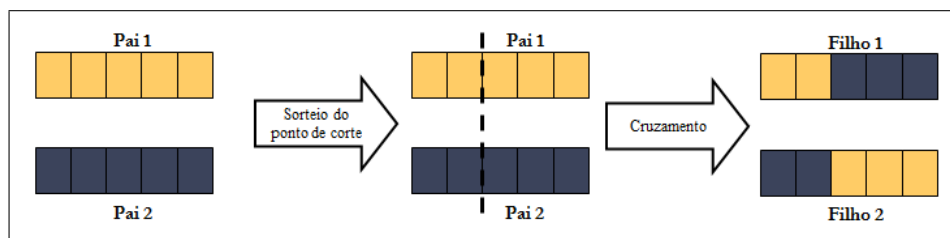


Figura 9: Cruzamento de 1 ponto [17]

3.5.2 Cruzamento de 2 Pontos

O funcionamento dessa operação é similar ao cruzamento de 1 ponto, com um pequeno acréscimo: ao invés de sortear um ponto de corte, dois pontos de corte são sorteados. O primeiro filho será formado pela parte do primeiro pai fora dos pontos de corte e pela parte do segundo pai entre os pontos de corte e o segundo filho será formado pelas partes restantes [17]. A Figura 10 exemplifica essa forma de cruzamento.

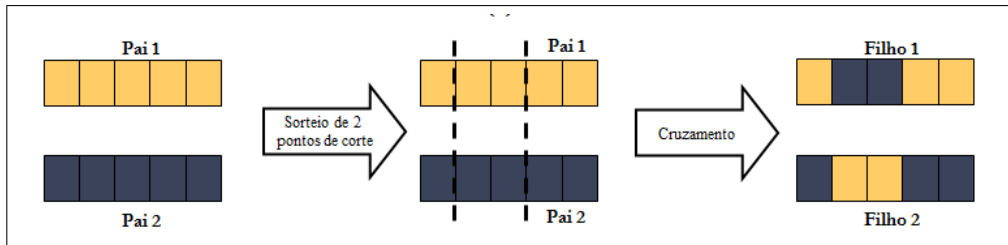


Figura 10: Cruzamento de 2 pontos [17]

3.5.3 Cruzamento Heurístico Uniforme

No cruzamento heurístico uniforme, cada par de genes correspondentes nos dois pais são comparados em termos do seu *fitness* local. O gene com maior *fitness* é copiado para um dos filhos enquanto o outro gene é copiado para o segundo filho [11]. A Figura 11 exemplifica essa forma de cruzamento.

Pai 1									
0	15	7	2	20	13	9	0	5	9
Pai 2									
4	0	12	1	8	25	2	7	0	8
Filho 1									
4	15	12	2	20	25	9	7	5	9
Filho 2									
0	0	7	1	8	13	2	0	0	8

Figura 11: Cruzamento heurístico uniforme

3.5.4 Cruzamento Heurístico Não Uniforme

No cruzamento heurístico não uniforme, é feita a cópia da informação genética dos pais para os filhos. Na sequência, deve-se identificar um número de genes (um parâmetro normalmente definido como $1/5$ do tamanho do cromossomo) com menores *fitness* locais em cada filho. Então, para cada filho, esses genes com menores *fitness* são copiados para o outro filho [11]. A Figura 12 exemplifica essa forma de cruzamento.

Pai 1									
0	15	7	2	20	13	9	0	5	9
Pai 2									
4	0	12	1	8	25	2	7	0	8
Filho 1									
0	15	7	2	20	13	9	0	5	9
Filho 2									
4	0	12	1	8	25	2	7	0	8
Filho 1									
0	0	7	2	20	13	9	0	0	9
Filho 2									
0	0	12	1	8	25	2	0	0	8

Figura 12: Cruzamento heurístico não uniforme

3.6 Mutação

Em um AG simples a mutação é uma alteração aleatória ocasional (com pequena probabilidade) do valor de um gene. A mutação é necessária porque, apesar de a seleção e o cruzamento buscarem e recombinarem efetivamente, ocasionalmente, eles podem tornar-se super-zelosos e perderem algum material genético potencialmente útil. O operador de mutação protege contra essa perda irreversível [18].

A mutação ocorre para que os filhos possam ter características que não foram herdadas diretamente dos seus pais, garantindo assim uma maior diversidade da população gerada. Considerando um indivíduo com representação binária, a operação de mutação pode ser exemplificada pela Figura 13.

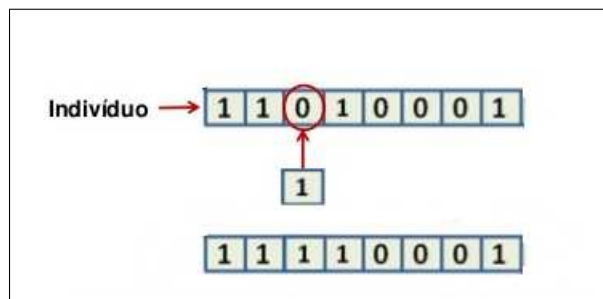


Figura 13: Mutação em um indivíduo com representação binária

A mutação em indivíduos com representação binária é a simples troca de um *bit* 0 por um *bit* 1, ou a troca de um *bit* 1 por um *bit* 0.

3.7 Escolha de Pais e Filhos Para Próxima Geração

Uma vez que os operadores de cruzamento e mutação tenham sido aplicados, é necessário decidir quais indivíduos daquela geração serão substituídos pela nova prole para formar a nova população. Uma estratégia não elitista substitui todos os indivíduos da população atual, enquanto uma estratégia elitista mantém os melhores indivíduos para

que seu material genético possa ser transferido para as próximas gerações [22].

O elitismo é uma pequena alteração no módulo de população que praticamente não altera o tempo de processamento de um AG. O elitismo garante que o desempenho do AG sempre cresça com o decorrer das gerações [17].

A ideia básica do elitismo é que os n melhores indivíduos de cada geração não devem morrer, eles devem passar para a próxima geração para garantir que seus genomas sejam preservados [17].

A manutenção do melhor indivíduo da geração t na população da geração $t+1$ garante que o melhor indivíduo da geração $t+1$ é pelo menos igual que o melhor indivíduo da geração t [17].

A curva de desempenho típica de um problema de minimização em um AG com elitismo é indicada pela Figura 14.

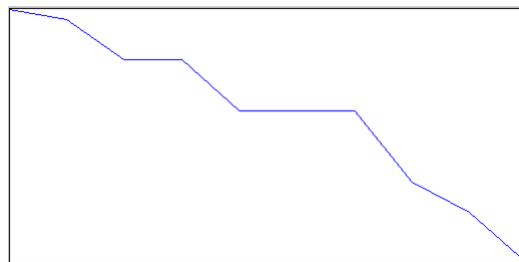


Figura 14: Curva típica de um AG com elitismo em um problema de minimização

3.8 Critérios de parada

Os operadores genéticos são aplicados iterativamente em um AG até que uma condição de parada seja alcançada. A condição de parada mais simples é limitar o número de gerações que um AG pode executar. O limite não deve ser muito pequeno, caso contrário, o AG não terá tempo suficiente para explorar o espaço de busca [20].

Além de limitar a quantidade de gerações, um critério de convergência é utilizado

para detectar se a população convergiu. A convergência é vagamente definida como um evento em que a população se torna estagnada. Os seguintes critérios de convergência podem ser utilizados [20]:

- Terminar quando nenhuma melhora é observada em um certo número consecutivo de gerações;
- Terminar quando não existe mudança no *fitness* médio da população;
- Terminar quando uma solução aceitável for encontrada;

Capítulo 4

Materiais e Métodos

4.1 Funcionamento do Algoritmo Genético

O PAS é um problema de otimização combinatória que é caracterizado pelas suas restrições. Para Engel em [20], as restrições colocam limitações no espaço de busca, especificando regiões no espaço que contêm soluções não admissíveis. O objetivo dos algoritmos de otimização nesses casos é encontrar soluções que estejam em regiões admissíveis.

Em problemas com restrições, a utilização de operadores genéticos nos indivíduos faz com que seja muito provável a criação de soluções não admissíveis. Técnicas de tratamento de restrições para algoritmos genéticos podem ser agrupadas em três categorias [19]:

1. Permitir a violação de restrições, mas penalizá-las;
2. Aplicar heurísticas especiais de reparação para corrigir as soluções não admissíveis;
3. Utilizar representações especiais de indivíduos para garantir ou aumentar a probabilidade de geração de apenas soluções admissíveis ou utilizar operadores específicos para o problema que preservem a viabilidade das soluções.

Considerando os requisitos computacionais da UFU indicados no Capítulo 2, verifica-se que o requisito ‘Utilizar a relação de cursos e blocos para a alocação de espaço físico’ não é uma restrição para o AG. De acordo com as técnicas para tratamento de restrições listadas, foi determinado que o indivíduo tenha uma representação especial e que

as operações de inicialização, cruzamento e mutação sejam específicas para o problema. A Tabela 4 indica como é feito o tratamento das restrições para o problema da UFU.

Tabela 4: Tratamento de restrições para o problema da UFU

Restrição	Tratamento
Alocar todas as turmas ofertadas que indicarem a necessidade de alocação de sala	Representação do indivíduo
Alocar turmas do mesmo curso e período em uma única sala quando os horários estiverem encaixados	Representação do indivíduo e operações genéticas específicas
Em uma mesma sala, horário e período não pode haver mais de uma turma	Função de avaliação
Uma sala não pode receber uma turma cuja quantidade de alunos seja superior a sua capacidade	Inicialização da população e mutação - lógica condicional
Turmas sem necessidade de prancheta devem ser alocadas apenas em salas com carteiras; turmas com necessidade prancheta devem ser alocadas apenas em salas com pranchetas	Inicialização da população e mutação - lógica condicional

O indivíduo do problema é representado por um cromossomo de tamanho igual à quantidade de turmas que devem ser direcionadas a um determinado bloco da universidade, e cada turma é representada por um gene do cromossomo. Os valores que os

genes podem assumir são do tipo inteiro e correspondem a um número de identificação (chave primária da tabela) das salas disponíveis para alocação no bloco.

Para atender à restrição de alocação das turmas de um mesmo curso e período na mesma sala, os genes são agrupados em seções de forma que os genes dentro de uma mesma seção conttenham o mesmo valor de sala.

A montagem do cromossomo e suas seções é feita de forma automática. Para isso é utilizado a relação de cursos e blocos e as informações de curso ofertante, período ideal e necessidade de alocação de cada turma. Além disso, o administrador de espaço físico pode manter as seções manualmente conforme desejar.

Conforme apresentado no Capítulo 2, cada turma possui um conjunto de informações vinculadas a ela. Essas informações são essenciais nas etapas de processamento do AG. A Figura 15 ilustra um indivíduo com quantidade de turmas igual a x e com quantidade de seções igual a y .

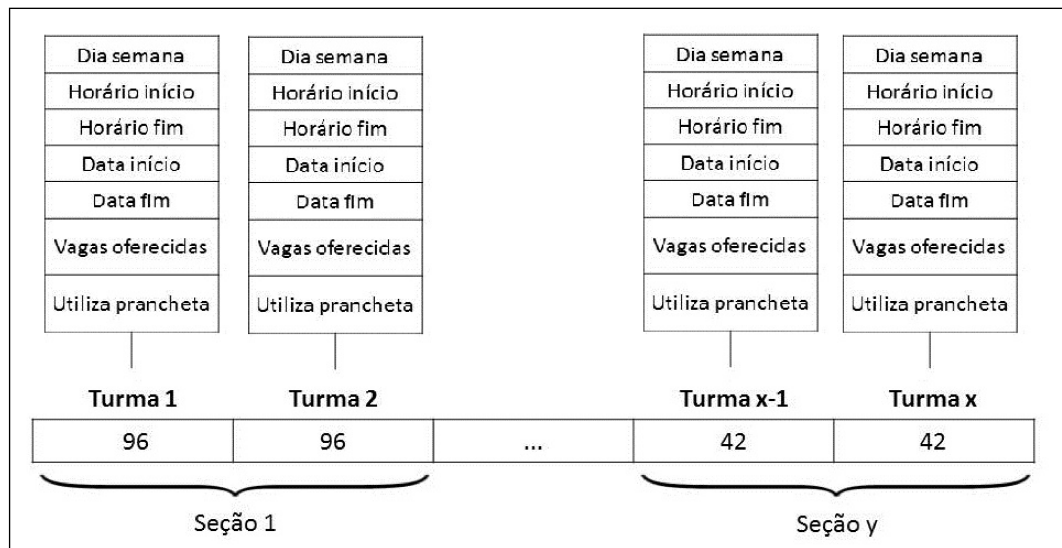


Figura 15: Representação do indivíduo

A Figura 16 apresenta uma pequena parte de um indivíduo com dados reais.

Dia semana	5	3	4	3	6	5	2	
Horário Início	14:00:00	08:00:00	08:00:00	14:00:00	08:00:00	08:00:00	08:00:00	
Horário Fim	17:40:00	11:30:00	11:30:00	17:40:00	11:30:00	11:30:00	11:30:00	
Data início	29/02/2016	29/02/2016	29/02/2016	29/02/2016	29/02/2016	29/02/2016	29/02/2016	
Data fim	02/07/2016	02/07/2016	02/07/2016	02/07/2016	02/07/2016	02/07/2016	02/07/2016	
Vagas oferecidas	42	49	50	45	50	47	46	
Utiliza Prancheta	N	N	N	N	N	N	N	
Indivíduo	3672	3670	3670	3670	3670	3670	3670	...
Seção	1	2						

Figura 16: Parte de um indivíduo com dados reais

A inicialização da população para o problema em questão é feita de forma aleatória considerando a representação do indivíduo definida. Para cada cromossomo e para cada seção do cromossomo, uma sala é sorteada, e se esta não violar as restrições de vagas e utilização de prancheta, então essa sala é atribuída a todos os genes da seção, caso contrário, o sorteio continua até encontrar uma sala adequada.

Da forma como a inicialização da população é realizada, é inevitável que choques de alocação (turmas alocadas na mesma sala, dia, horário e período de realização) não aconteçam quando a razão entre o número de turmas e a quantidade de salas é grande. Portanto, os indivíduos serão avaliados pela quantidade de choques de alocação que os mesmos possuem.

O objetivo do algoritmo genético é minimizar a quantidade de choques de alocação até encontrar um indivíduo que não possua choques, ou seja, que possui avaliação igual à zero. Um indivíduo com avaliação igual à zero representa uma das possíveis soluções desejadas. Considerando q a quantidade de choques de alocação, a função de avaliação (*fitness*) do indivíduo é definida como:

$$fitness = \sum q \quad (5)$$

O método de seleção escolhido foi o torneio. Nenhuma modificação é necessária no método devido à representação do indivíduo proposta.

A forma de cruzamento escolhida para este trabalho foi o heurístico não uniforme. Considerando o indivíduo proposto, ao invés de identificar os genes com menores avaliações, são identificadas as seções com menores médias de avaliação. O valor médio de avaliação é utilizado, pois as seções podem conter diferentes quantidades de genes, caso fosse utilizado a soma das avaliações, as seções com menor quantidade de genes levariam vantagem sobre as seções com maior quantidade de genes.

Na Figura 17 é demonstrado o cruzamento heurístico não uniforme para um cromossomo com 20 genes distribuídos em 8 seções, onde as diferentes cores indicam cada uma das seções. Foram destacadas as duas seções com menores valores médios em cada filho. Os números em cada gene representam sua avaliação ao invés do valor de uma sala para facilitar a compreensão da operação.

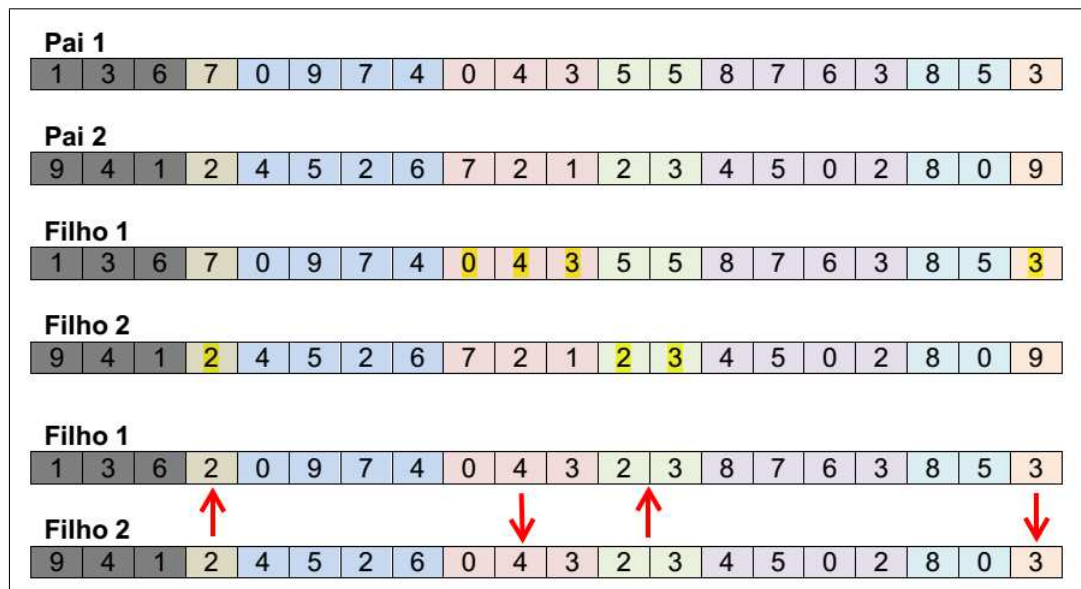


Figura 17: Cruzamento heurístico não uniforme para o indivíduo proposto

Uma vez que a inicialização da população garante indivíduos que não violam as

restrições de vagas e utilização de pranchetas, os indivíduos resultantes do cruzamento também não violam as restrições. Isso é uma grande vantagem, pois não é necessária a utilização de rotinas para reparar os indivíduos e satisfazer as restrições.

Na mutação, ao invés dela ocorrer nos genes, ela ocorre nas seções. Quando uma determinada seção é sorteada para mutação, todos os genes dessa seção são modificados. Caso o novo valor sorteado viole as restrições de vagas e utilização de pranchetas, novos sorteios são feitos até ser encontrado um valor que não as viole. A Figura 18 exemplifica essa operação para o indivíduo proposto.

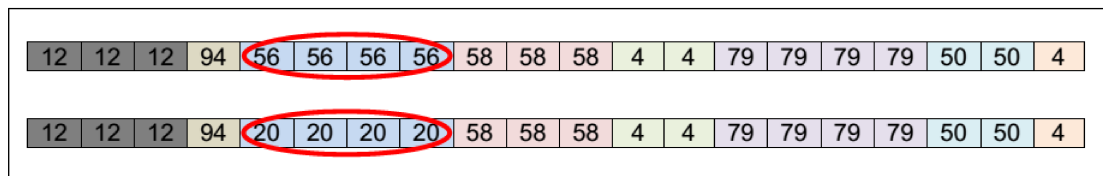


Figura 18: Mutação para o indivíduo proposto

Quanto a escolha de pais e filhos para próxima geração, é utilizada a estratégia elitista com 1 indivíduo. O pior indivíduo da nova geração é substituído pelo melhor indivíduo da geração passada, garantindo que o melhor indivíduo da nova geração tenha avaliação pelo menos igual ao melhor indivíduo da geração passada.

Os critérios de parada definidos para o AG deste trabalho são os seguintes:

1. Encontrar um indivíduo admissível (com avaliação igual à 0), pois qualquer indivíduo admissível é uma resposta ao PAS;
2. Máximo de 500 gerações consecutivas com o melhor indivíduo apresentando o mesmo valor;
3. Máximo de 2000 gerações.

Os parâmetros para execução do algoritmo genético estão definidos na Tabela 5.

Tabela 5: Parâmetros de execução do AG

Parâmetros	Valor
Qtd. indivíduos da população	200
Qtd. indivíduos no torneio	10
Taxa de cruzamento	0,7
Taxa de mutação	0,05

4.2 Tecnologias Empregadas

Na UFU, o Sistema de Gestão (SG) é responsável por apoiar diversas áreas dentro da instituição. Os usuários podem acessar o SG pelo sítio www.sg.ufu.br com sua devida autenticação. O SG possui aplicações relacionadas ao controle de frota, solicitação de material e compras, acesso ao restaurante universitário, controle acadêmico entre outras.

A oferta de disciplinas é realizada pelo SG, portanto o desenvolvimento de aplicações para alocação de espaço físico nesse sistema possibilita que os dados sejam utilizados de forma integrada.

Na parte de visão são utilizadas as tecnologias: HTML, CSS e JavaScript. Para a parte de controle é utilizado Java versão 7. O banco de dados utilizado é o IBM DB2 versão 9.7. A IDE de desenvolvimento utilizada é o Eclipse Mars. Foi utilizado um computador com processador Intel Core i7-3770 3.40 GHz e 8 GB de memória.

4.3 Aplicações Desenvolvidas como Resultado deste Trabalho

A aplicação 11.02.03.11 é responsável por manter a relação de prédios e cursos. Nela, o usuário informa um prédio e, em seguida, adiciona ou exclui os períodos de um curso que terão suas turmas alocadas no prédio informado. Inicialmente foi feita uma

importação dos dados dessa relação que estavam disponíveis em uma planilha. Para o processamento do 2º semestre de 2016 não foram necessárias modificações. A Figura 19 apresenta a interface dessa aplicação.

11.02.03.11 - ALOCAÇÃO ESPAÇO FÍSICO - RELAÇÃO DE PRÉDIOS E CURSOS

Bloco
50-A

Cursos e Períodos Disponíveis para Alocação

Cursos **Período Ideal**

RELAÇÃO PRÉDIOS CURSOS

Curso	Período Ideal
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	0
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	1
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	2
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	3
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	4
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	5
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	6
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	7
0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	8

Figura 19: Aplicação 11.02.03.11 - Relação de prédios e cursos

A aplicação 11.02.03.12 é responsável por manter a relação de turmas e seções. Nela, o usuário possui diferentes opções para gerenciamento das seções, além de possuir relatórios de apoio.

Na aba ‘Cadastro’, o usuário pode adicionar, alterar e excluir seções e turmas. Quando uma seção está selecionada, a relação das turmas vinculadas à ela é renderizada na tela em forma de lista. Dessa forma o usuário pode visualizar todas as informações das turmas dentro de uma seção. A Figura 20 apresenta a aba ‘Cadastro’.

11.02.03.12 - ALOCAÇÃO ESPAÇO FÍSICO - RELAÇÃO DE TURMAS E SEÇÕES

Cadastro | Geração Automática | Relatório

Bloco: 50-A Ano: 2016 Período: 2º Semestre

Seções: Graduação em Ciências Sociais - Licenciatura - Diurno - 4º Período

Descrição Seção:

TURMAS

	Curso	Disciplina	Período	Dia	Horário	Início/Fim Período	Tipo	V/MO	Prancheta
☐	0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	GCS019 - Antropologia IV - S	4	Sexta-feira	08:00:00 às 11:30:00	08/08/2016 à 17/12/2016	Teórica	60/26	N
☐	0344LCCS - Graduação em Ciências Sociais - Licenciatura - Diurno	GCS020 - Ciência Política IV - S	4	Quinta-feira	08:00:00 às 11:30:00	08/08/2016 à 17/12/2016	Teórica	60/28	N

Total de turmas: 6

ALTERAR **EXCLUIR**

Figura 20: Aplicação 11.02.03.12 - Relação de turmas e seções aba ‘Cadastro’

Na aba ‘Geração Automática’, o usuário possui duas opções para a criação de seções automaticamente. Caso opte pela opção ‘Relação Prédios Curso’, seções são criadas utilizando os dados da Relação de Prédios e Cursos e da Oferta de Disciplinas para o ano e semestre informado. A opção ‘Cópia’ realiza a cópia das seções utilizando o ano e período de referência indicados, além dos dados da Oferta de Disciplinas. A opção ‘Cópia’ ainda não foi utilizada, uma vez que ainda não existem dados para serem utilizados como referência. Após a geração automática das seções, o usuário ainda pode modificá-las conforme desejar pela aba ‘Cadastro’. A Figura 21 apresenta a aba ‘Geração Automática’.

Figura 21: Aplicação 11.02.03.12 - Relação de turmas e seções aba ‘Geração Automática’

Na aba ‘Relatório’, o usuário possui a opção de 3 relatórios. A Figura 22 apresenta a aba ‘Relatório’.

Figura 22: Aplicação 11.02.03.12 - Relação de turmas e seções aba ‘Relatório’

O relatório ‘Relação Turmas e Seções’ apresenta as informações das seções criadas para um determinado bloco, ano e período. Ele fornece uma visão mais generalizada em relação à aba ‘Cadastro’. A Figura 23 apresenta uma página deste relatório.

 UNIVERSIDADE FEDERAL DE UBERLÂNDIA 11.02.03.12 Relação de Turmas e Seções Data: 23/09/2016 Hora: 10:31											
Prédio: 50-A Ano: 2016 Período: 2º Semestre											
Seção: Graduação em Ciências Sociais - Licenciatura - Diurno - 4º Período											
Curso	Disciplina	Período Ideal	Dia Semanal	Horário		Data		Tipo	Vagas/MD	Prancheta	
				Início	Fim	Início	Fim	Aula			
0344LCCB - Graduação em Ciências Sociais -	GO8019 - Antropologia IV - 8	4	Sexta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/26	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8020 - Ciência Política IV - 8	4	Quinta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/28	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8021 - Sociologia IV - 8	4	Quarta-feira	07:10	10:40	08/08/2016	17/12/2016	Teórica	60/29	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8022 - Métodos e Técnicas de Pesquisa III - 8	4	Terça-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/24	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8023 - Introdução à Estatística - 8	4	Segunda-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/31	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8024 - Metodologia de Ensino em Ciências Sociais	4	Quarta-feira	10:40	12:20	08/08/2016	17/12/2016	Teórica	60/22	Não	
Seção: Graduação em Ciências Sociais - Licenciatura - Diurno - 6º Período											
Curso	Disciplina	Período Ideal	Dia Semanal	Horário		Data		Tipo	Vagas/MD	Prancheta	
				Início	Fim	Início	Fim	Aula			
0344LCCB - Graduação em Ciências Sociais -	GO8031 - Antropologia no Brasil - 8	6	Quinta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/24	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8032 - Pensamento Sociológico Brasileiro - 8	6	Segunda-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/30	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8033 - Política Brasileira II - 8	6	Sexta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/24	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8035 - Estágio Supervisionado II - 8	6	Terça-feira	13:10	17:40	08/08/2016	17/12/2016	Teórica	60/17	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8035 - Estágio Supervisionado II - 8	6	Quarta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/17	Não	
Seção: Graduação em Ciências Sociais - Licenciatura - Diurno - 8º Período											
Curso	Disciplina	Período Ideal	Dia Semanal	Horário		Data		Tipo	Vagas/MD	Prancheta	
				Início	Fim	Início	Fim	Aula			
0344LCCB - Graduação em Ciências Sociais -	GO8041 - Tópicos Especiais em Ciências Sociais II -	8	Segunda-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/12	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8042 - Análise Política Contemporânea - 8	8	Sexta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/32	Não	
0344LCCB - Graduação em Ciências Sociais -	GO8044 - Estágio Supervisionado IV - 8	8	Quarta-feira	08:00	11:30	08/08/2016	17/12/2016	Teórica	60/18	Não	
0344LCCB - Graduação em Ciências Sociais -	LIBRAS01 - Língua Brasileira de Sinais - Libras I - V/I	8	Sexta-feira	14:00	17:40	08/08/2016	17/12/2016	Teórica	51/24	Não	
Seção: Graduação em Filosofia - Licenciatura - Noturno - 2º Período TURMAS GRANDES											
Curso	Disciplina	Período Ideal	Dia Semanal	Horário		Data		Tipo	Vagas/MD	Prancheta	
				Início	Fim	Início	Fim	Aula			
0358LFIL - Graduação em Filosofia - Licenciatura -	GF1007 - História da Filosofia Antiga 2 - F	2	Segunda-feira	19:00	22:30	08/08/2016	17/12/2016	Teórica	35/58	Não	
0358LFIL - Graduação em Filosofia - Licenciatura -	GF1008 - Filosofia Geral: Problemas Metafísicos 2 - F	2	Quinta-feira	19:00	22:30	08/08/2016	17/12/2016	Teórica	35/54	Não	
0358LFIL - Graduação em Filosofia - Licenciatura -	GF1010 - Leitura e Produção de Textos em Filosofia 2	2	Sexta-feira	19:00	22:30	08/08/2016	17/12/2016	Teórica	70/57	Não	
0358LFIL - Graduação em Filosofia - Licenciatura -	GF1011 - Projetos Interdisciplinares de Pesquisa e	2	Sábado	08:00	09:40	08/08/2016	17/12/2016	Prática	70/55	Não	

Figura 23: Relatório 11.02.03.12 - Relação turmas seções

O relatório ‘Seções Críticas’ apresenta as seções que possuem turmas muito divergentes em relação à quantidade de vagas. Sua importância reside no fato de que apenas uma sala que comporte a maior turma de uma seção pode ser alocada para todas as turmas dessa seção. Diante disso, o usuário pode remanejar as turmas para seções que possuem turmas com quantidade de vagas mais próximas. A Figura 24 apresenta este relatório.

		UNIVERSIDADE FEDERAL DE UBERLÂNDIA		Data: 23/09/2016	
		11.02.03.12 Relação de Turmas e Seções - Seções Críticas		Hora: 10:31	
Campus: Campus Santa Mônica		Prédio: 5O-A		Ano/Período: 2016/2º Semestre	
Seção				Vagas	
				Mínimo	Máximo
Curs.de Lic.Pl.em Letras-H.em Port. e L.de L. Port. - Diurno - 4º Período				35	60
Curso de Graduação em História: Bacharelado e Licenciatura - Diurno e Noturno - 6º Período				21	40
Curso de Graduação em História: Bacharelado e Licenciatura - Diurno e Noturno - 8º Período				21	40
Curso de Lic.Plena em Letras-H. em Port.e L. de L.Port. Not. - 4º Período				30	50
Curso de Licenciatura em Geografia - Diurno e Noturno - 2º Período				50	65
Curso de Licenciatura em Geografia - Diurno e Noturno - 6º Período				29	50
Curso de Licenciatura em Geografia - Diurno e Noturno - 8º Período				10	50
Graduação em Filosofia - Licenciatura - Noturno - 2º Período TURMAS GRANDES				35	70
Graduação em Filosofia - Licenciatura - Matutino e Noturno - 2º Período				35	70
Graduação em Filosofia - Licenciatura - Matutino e Noturno - 6º Período				40	90
Graduação em Filosofia - Licenciatura - Matutino e Noturno - 8º Período				40	90
Graduação em Química : Licenciatura - Noturno - 2º Período				20	80
Graduação em Química : Licenciatura - Noturno - 4º Período				40	80

Figura 24: Relatório 11.02.03.12 - Seções críticas

O relatório 'Qtd. de Turmas por Horário' apresenta um mapa da quantidade de turmas existentes para um determinado bloco, ano e período em cada horário de aula de uma semana. Com este relatório, o usuário pode visualizar os horários que possuem maior concentração de turmas. Caso um horário possua quantidade de turmas maior que a quantidade de salas do bloco, é certo que o processamento terminará com choques de alocação. A Figura 25 apresenta este relatório.

 Universidade Federal de Uberlândia 11.02.03.12 - Quantidade de Turmas por Horário Campus Santa Mônica Prédio: 50-A Qtd. Salas: 26 Data: 23/09/2016 Hora: 10:31 2º Semestre/2016						
HORÁRIO	2 Segunda - feira	3 Terça - feira	4 Quarta - feira	5 Quinta - feira	6 Sexta - feira	7 Sábado
1 07:10 às 08:00			2			
2 08:00 às 08:50	18	25	25	21	19	4
3 08:50 às 09:40	20	26	25	23	19	4
4 09:50 às 10:40	20	26	25	23	19	3
5 10:40 às 11:30	20	26	25	23	19	3
6 11:30 às 12:20	6	7	5	6	4	
7 13:10 às 14:00	2	2	3	4	3	
8 14:00 às 14:50	1	5	1	4	3	4
9 14:50 às 15:40	1	5			1	4
10 16:00 às 16:50	1	5	2	4	3	3
11 16:50 às 17:40	1	4	3	4	5	3
12 17:40 às 18:30			2		1	
13 18:10 às 19:00	3	6	4	6		
14 19:00 às 19:50	17	22	18	20	18	
15 19:50 às 20:40	17	22	18	20	18	
16 20:50 às 21:40	17	21	18	21	18	
17 21:40 às 22:30	17	20	18	20	17	

Figura 25: Relatório 11.02.03.12 - Quantidade de turmas por horário

Após a montagem das seções, o processamento da alocação de salas é feito na aplicação 11.02.03.13. Basta que o usuário informe o bloco, ano e período e clique no botão ‘Processar’. Uma vez iniciado o processamento, o usuário pode acompanhar o progresso pela busca da solução por uma janela que é atualizada a cada segundo. A Figura 26 apresenta essa aplicação no início do processamento e próximo a encontrar uma solução admissível.

11.02.03.13 - ALOCAÇÃO ESPAÇO FÍSICO - PROCESSAMENTO ALOCAÇÃO TEMPORÁRIA

Bloco
Ano
Período
50-A
2016
2º Período
PROCESSAR

Processando ...

Bloco: 50-A Ano: 2016 Período: 2º Período

Início: 23/09/2016 11:05:14
Quantidade de Gerações: 0
Quantidade de choques: 183
Última atualização: 11:05:21

11.02.03.13 - ALOCAÇÃO ESPAÇO FÍSICO - PROCESSAMENTO ALOCAÇÃO TEMPORÁRIA

Bloco
Ano
Período
50-A
2016
2º Período
PROCESSAR

Processando ...

Bloco: 50-A Ano: 2016 Período: 2º Período

Início: 23/09/2016 11:05:14
Quantidade de Gerações: 65
Quantidade de choques: 2
Última atualização: 11:05:47

Figura 26: Aplicação 11.02.03.13 - Processamento alocação temporária

Finalizado o processamento, o usuário verifica o resultado na aplicação 11.02.03.14. Nela, é possível visualizar as turmas que foram alocadas em cada sala em forma de grade horária. A Figura 27 apresenta essa aplicação.

11.02.03.14 - ALOCAÇÃO ESPAÇO FÍSICO - ALOCAÇÃO TEMPORÁRIA

Ano

2016

Período

2º Período

Bloco

50-A

Sala

50-A 204

Data Início/Fim Período

Todos

Sala: 50-A 204

Descrição: Plana com carteiras

Capacidade: 42

Horário	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Legenda
07:10 - 08:00			GCS021-S				
08:00 - 08:50	GCS023-S	GCS022-S	GCS021-S	GCS020-S	GCS019-S	GQL040-Q	GLE010 - Introdução aos Estudos da Linguagem
08:50 - 09:40	GCS023-S	GCS022-S	GCS021-S	GCS020-S	GCS019-S	GQL040-Q	Graduação em Ciências Sociais - Licenciatura - Diurno - 4º Período
09:50 - 10:40	GCS023-S	GCS022-S	GCS021-S	GCS020-S	GCS019-S	GQL040-Q	GCS019 - Antropologia IV
10:40 - 11:30	GCS023-S	GCS022-S	GCS024-S	GCS020-S	GCS019-S	GQL040-Q	GCS020 - Ciência Política IV
11:30 - 12:20			GCS024-S				GCS021 - Sociologia IV
13:10 - 14:00							GCS022 - Métodos e Técnicas de Pesquisa III
14:00 - 14:50	IGUFU338		GCS025-S	IGUFU338	IGUFU338	GQL040-Q	GCS023 - Introdução à Estatística
14:50 - 15:40	IGUFU338		GCS025-S	IGUFU338	IGUFU338	GQL040-Q	GCS024 - Metodologia de Ensino em Ciências Sociais IV
16:00 - 16:50	IGUFU338		GCS025-S	IGUFU338	IGUFU338	GQL040-Q	Graduação em Química : Licenciatura - Noturno - 10º Período
16:50 - 17:40	IGUFU338			IGUFU338	IGUFU338	GQL040-Q	GQL041 - Química Aplicada 2
17:40 - 18:30							Turmas sem seção
18:10 - 19:00				GQL040-Q	GQL040-Q		GCS025 - Projeto Integrado de Prática Educativa IV
19:00 - 19:50		GLE010-N	GQL040-Q				GQL040 - Estágio Supervisionado 4
19:50 - 20:40		GLE010-N	GQL040-Q				GQL061 - Química Quântica
20:50 - 21:40		GLE010-N	GQL061-Q	GQL041-Q			IGUFU33801 - Educação Ambiental
21:40 - 22:30		GLE010-N	GQL061-Q	GQL041-Q			IGUFU33802 - Planejamento Ambiental
							IGUFU33803 - Vigilância Epidemiológica

Horário possui choque de disciplinas.

Horário de turmas anuais.

Choques de Alocação

Figura 27: Aplicação 11.02.03.14 - Alocação temporária

Caso o processamento seja finalizado com choques de alocação, os mesmos são destacados na cor vermelha nos horários em que eles ocorreram. A Figura 28 apresenta essa funcionalidade.

11.02.03.14 - ALOCAÇÃO ESPAÇO FÍSICO - ALOCAÇÃO TEMPORÁRIA

LIMPAR

Imprimir

Salvar

PDF

Ano

2016

Período

1º Período

Bloco

1B

Sala

1B 104

Data Início/Fim Período

Todos

Sala: 1B 104

Descrição: Anfiteatro com carteiras

Capacidade: 80

Horário	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Legenda
07:10 - 08:00	FAMAT490	INFIS490	FAMAT490				Graduação em Engenharia Mecatrônica - Bacharelado - 3º Período
08:00 - 08:50	FAMAT490	INFIS490	FAMAT490		FEMEC420		FAMAT49030 - Cálculo Diferencial e Integral III
08:50 - 09:40	FECIV490	FAMAT490	INFIS490	FECIV490	FEMEC420		FECIV49032 - Estática
09:50 - 10:40	FECIV490	FAMAT490	INFIS490	FECIV490	FEMEC420		FEMEC41030 - Cinemática
10:40 - 11:30	INFIS490				FEMEC420		FEMEC42031 - Introdução à Ciências dos Materiais
11:30 - 12:20	INFIS490				FEMEC420		FEMEC42032 - Projeto Assistido por Computador
13:10 - 14:00		FAMAT490	FAMAT490	INFIS490	FAMAT490		INFIS49030 - Física Geral II
14:00 - 14:50		FAMAT490	FAMAT490	INFIS490	FAMAT490		Graduação em Engenharia Mecânica - Bacharelado - 2º Período
14:50 - 15:40	FAMAT490	FAMAT490	INFIS490	FAMAT490			FAMAT49020 - Cálculo Diferencial e Integral II
16:00 - 16:50	FAMAT490	FAMAT490	INFIS490	FAMAT490			FAMAT49021 - Estatística
16:50 - 17:40		FAMAT490					FAMAT49022 - Álgebra Linear
17:40 - 18:30							INFIS49020 - Física Geral I
18:10 - 19:00							Graduação em Sistemas de Informação - 3º Período
19:00 - 19:50	GSIO15-S	GSIO11-S	GSIO15-S	GSIO13-S	GSIO12-S		GSIO11 - Estrutura de Dados 2
19:50 - 20:40	GSIO15-S	GSIO11-S	GSIO15-S	GSIO13-S	GSIO12-S		GSIO12 - Matemática 2
20:50 - 21:40	GSIO13-S	GSIO12-S	GSIO14-S	GSIO11-S	GSIO14-S		GSIO13 - Arquitetura e Organização de Computadores
21:40 - 22:30	GSIO13-S	GSIO12-S	GSIO14-S	GSIO11-S	GSIO14-S		GSIO14 - Matemática para Ciência da Computação
							GSIO15 - Programação Orientada a Objetos 1

Horário possui choque de disciplinas.

Horário de turmas anuais.

Choques de Alocação

Figura 28: Aplicação 11.02.03.14 - Choques de alocação destacados na cor vermelha

Ao clicar em algum horário com disciplina alocada na grade, o usuário possui as opções de realocar ou excluir o mesmo. Essa funcionalidade permite que o usuário modifique o resultado do processamento conforme desejar, possibilitando a eliminação dos choques de alocação quando existentes. A Figura 29 apresenta essa funcionalidade.

11.02.03.14 - ALOCAÇÃO ESPAÇO FÍSICO - ALOCAÇÃO TEMPORÁRIA

Exclusão e Realocação de Turmas

Bloco 1B Sala: 1B 104 - 2016/1º Período

TURMAS SELECIONADAS

	Curso	Disciplina	Dia	Horário	Início/Fim Período	Tipo	V/MO	Prancheta
☐	1450BI - Graduação em Engenharia Mecânica: Bacharelado - Integral	FAMAT49020 - Cálculo Diferencial e Integral II - U	Terça-feira	13:10:00 às 14:50:00	29/02/2016 à 02/07/2016	Teórica	70/46	N
☐	71863BI - Graduação em Engenharia Mecatrônica: Bacharelado - Integral	FEMEC41030 - Cinemática - V	Terça-feira	13:10:00 às 15:40:00	29/02/2016 à 02/07/2016	Teórica	35/32	N

AÇÃO

Opção: Bloco Destino: Sala Destino: Sala: Descrição: Capacidade:

Realocação
Excluir

Figura 29: Aplicação 11.02.03.14 - Exclusão e realocação de turmas

O passo final do processo consiste em alocar definitivamente as salas para as turmas a partir da alocação temporária. A janela 'Alocação Definitiva' ao ser aberta verifica pela existência de choques de alocação no prédio e informa ao usuário em quais salas eles ocorrem. Caso o prédio não possua choques de alocação, a aplicação habilita o botão 'Confirmar'. A Figura 30 apresenta essa funcionalidade.

Capítulo 5

Resultados e Discussões

A primeira etapa deste capítulo é testar o algoritmo genético proposto e verificar a viabilidade utilizando o resultado da alocação no bloco 5O-A do primeiro semestre de 2016. O processo de alocação de espaço físico no primeiro semestre de 2016 foi feito manualmente, portanto, as restrições foram atendidas e na solução adotada não existem choques de alocação. Sendo assim, espera-se que o AG faça a redistribuição de salas e que encontre um indivíduo sem choques de alocação.

No bloco 5O-A foram alocadas 420 turmas distribuídas em 26 salas de aula. As 420 turmas foram divididas em 88 seções. O bloco possui apenas salas com carteira com quatro tipos diferentes de capacidade (40, 42, 45 e 50).

A Tabela 6 apresenta os resultados obtidos em 10 execuções do AG. A coluna ‘Inicial’ indica qual a avaliação do melhor indivíduo após a inicialização da população. A coluna ‘Final’ indica qual a avaliação do melhor indivíduo na última geração. A coluna ‘Gerações’ indica a quantidade de gerações executadas quando o processamento foi interrompido.

Tabela 6: Resultados de execução bloco 5O-A

Execução	Qtd. de choques		Gerações	Tempos[s]
	Inicial	Final		
1	212	2	626	462
2	188	0	104	82
3	210	0	164	127
4	212	2	666	484
5	220	2	582	432
6	206	0	35	33
7	226	0	123	96
8	216	0	302	222
9	188	2	631	477
10	204	0	479	352

O comportamento da execução 2 é apresentado na Figura 31. Nessa figura é mostrada a avaliação do melhor indivíduo (curva vermelha) e também a média de avaliação de todos os indivíduos da população (curva azul) ao longo das gerações.

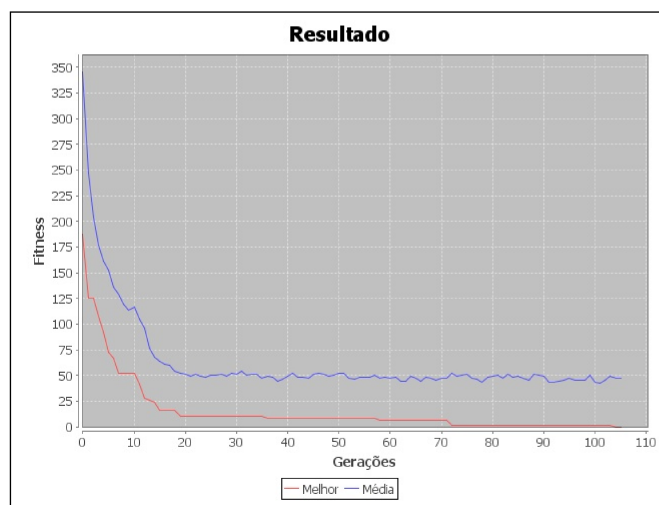


Figura 31: Resultado da execução 2

Pela Figura 31 observa-se que, nas gerações iniciais, a avaliação do melhor indivíduo sempre apresenta decaimento e quanto mais o AG se aproxima da solução, maior é a dificuldade de se encontrar um indivíduo com melhor avaliação.

Os resultados das 10 execuções mostram que o AG encontrou uma solução sem choques de alocação em 60% casos, enquanto nos outros 40% a solução encontrada apresentou 2 choques, o que representa uma queda de aproximadamente 99% da quantidade de choques inicial. Mesmo quando a solução admissível não é encontrada, a baixa quantidade de choques de alocação faz com que ela seja passível de ser analisada e corrigida pelo administrador de espaço físico da instituição. Os tempos de execução, mesmo no pior caso (8 minutos e 4 segundos), são satisfatórios levando-se em conta que seria necessário pelo menos um dia para fazer a alocação manualmente.

Uma vez que os resultados obtidos mostram a viabilidade de aplicação deste método no problema, a segunda etapa deste capítulo apresenta os resultados obtidos na aplicação deste método para os dados resultantes da oferta de disciplinas do 2º semestre de 2016.

A Tabela 7 apresenta os resultados das 20 execuções realizadas para os 14 blocos da cidade de Uberlândia. O método não foi aplicado nos campi das cidades de Ituiutaba, Patos de Minas e Monte Carmelo. A letra C indica a quantidade de salas com carteira, enquanto a letra P indica a quantidade de salas com prancheta.

Tabela 7: Resultados de execução para o 2º semestre de 2016

Bloco	Qtd. de Salas		Tipos diferentes de sala	Qtd. de turmas	Qtd. de seções	Qtd. de choques				Gerações		Tempo[s]	
						Inicial		Final					
	C	P				μ	σ	μ	σ	μ	σ	μ	σ
1B	11	0	1	548	26	327,2	51,63	22	0	560,7	51,39	862,2	81,08
1BCG	12	0	2	58	13	0	0	0	0	0	0	4	0
1C	4	0	1	64	4	0	0	0	0	0	0	4	0
2C	5	0	1	99	13	74	6,59	56	0	550	77,18	105,7	13,61
3D	15	0	2	228	45	46,4	7,99	0	0	10,95	3,25	9,25	1,16
3Q	29	4	9	633	89	254,7	17,41	4,8	2,46	975,4	343,62	1099,1	378,91
4K	11	0	4	107	14	0	0	0	0	0	0	4	0
4L	5	0	1	46	12	2,1	1,52	0	0	0,95	0,69	4	0
5O-A	26	0	4	392	73	181,1	13,29	1,4	1,31	460,7	244,73	256,8	132,4
5O-B	0	8	1	42	11	0	0	0	0	0	0	4	0
5R-A	26	0	3	692	70	425,7	49,78	3,2	2,09	759	227,95	1006,5	294,07
5R-B	8	0	2	129	22	41,4	6,87	8	0	516,05	21,41	128,45	5,15
5S	33	0	1	191	36	2,6	1,6	0	0	1,1	0,72	5,15	0,37
8C	48	0	5	880	128	701,1	34,89	28,8	3	1111,2	344,64	2089,8	651,96

Analisando os resultados obtidos na Tabela 7 juntamente com a relação de turmas e seções construídas, o administrador de espaço físico da UFU verificou que, para os blocos nos quais não foram encontrados indivíduos admissíveis ou que não chegaram próximos de uma solução, realmente não existia uma solução possível. Isso é devido aos seguintes fatores:

- Seções com um número muito grande de turmas, pois essas seções podem ocupar uma grande parte dos horários disponíveis em uma sala, dificultando o compartilhamento da mesma com outra seção;
- Seções heterogêneas em relação à quantidade de vagas, ou seja, turmas com poucas vagas na mesma seção do que turmas com muitas vagas, fazendo com que todas as turmas da seção sejam alocadas em uma sala maior;

- Maior quantidade de turmas do que salas em um determinado horário;

Apesar de não encontrar uma solução admissível para todos os blocos, o administrador de espaço físico da instituição possui duas maneiras de resolver os problemas de choque de alocação resultantes, sendo elas: (1) remontar as seções críticas e realizar o processamento novamente, ou (2) eliminar os choques de alocação através da realocação de turmas em outras salas.

Capítulo 6

Relato do Administrador de Espaços da UFU

A aba ‘Cadastro’ da aplicação 11.02.03.12 foi utilizada quando havia necessidade de montar seções que não atendem a relação de prédios e cursos, servindo como alternativa para a geração automática, e isso ofereceu diferentes opções para a melhor distribuição das turmas nos blocos.

Os relatórios disponíveis na aplicação 11.02.03.12 foram de grande utilidade. Uma vez que para cada bloco existe uma grande quantidade de seções e os relatórios proporcionaram uma maneira mais fácil de encontrar informações sobre as turmas, além de indicarem as seções e horários que poderiam dificultar o processamento da alocação de salas.

A aplicação 11.02.03.14 foi essencial na etapa de alterações de salas e/ou horários que ocorreram posteriormente à alocação definitiva. Feito manualmente, era preciso registrar as alterações em uma planilha e na aplicação de Oferta de Disciplinas do SG. Comparando as duas formas, o trabalho pela aplicação 11.02.03.14 foi mais fácil, uma vez que tudo foi resolvido em um único lugar, sem a necessidade de lidar com outros documentos e aplicações.

No processo manual de alocação de salas, após a finalização do planejamento em uma planilha, era preciso acessar a aplicação de Oferta de Disciplinas, e alocar as salas turma por turma. A funcionalidade de alocação definitiva da aplicação 11.02.03.14 forneceu agilidade nesta etapa, uma vez que todo o planejamento de alocação temporário pode ser feito definitivo pelo botão ‘Confirmar’ da janela ‘Alocação Definitiva’.

Considerando que o processo de alocação de salas é feito uma vez por semestre, sendo que mais de 4000 turmas devem ser distribuídas em 245 salas de aula, o sistema

desenvolvido tornou o processo prático e dinâmico. A execução do processo manual era realizada por um responsável com a ajuda de até dois estagiários consumindo um tempo total de 640 horas, com a implantação do sistema proposto, as atividades foram feitas apenas pelo responsável e levou em torno de 160 horas para ser finalizada.

Capítulo 7

Conclusões

Este trabalho demonstrou que estratégias baseadas em algoritmos genéticos são viáveis para a resolução do problema de alocação de salas da Universidade Federal de Uberlândia.

A contribuição deste trabalho foi apresentar uma maneira diferente de tratar um problema de natureza multi-objetiva. Normalmente resolvido com uma função de avaliação que representa a soma ponderada de todas as restrições impostas, a função de avaliação deste trabalho lida apenas com a restrição de múltiplas turmas em um único horário, enquanto as outras são tratadas pela representação especial do indivíduo, dos operadores genéticos específicos e por regras condicionais.

A comparação dos resultados de alocação automatizada do bloco 5O-A no 1º semestre de 2016 com a solução encontrada manualmente garantiu a viabilidade de aplicação do método, uma vez que o AG encontrou a solução (ou próximo desta) em todos os casos com um tempo de processamento muito baixo.

Nos dados referentes à oferta de disciplinas do 2º semestre de 2016, as alocações nos blocos 1B, 2C e 8C tiveram as piores performances devido à fatores mencionados no Capítulo 6. Mesmo assim, analisando os choques de alocação juntamente com os relatórios de apoio da aplicação 11.02.03.12, o administrador de espaço físico pode refazer a divisão das seções e conseguir uma solução viável. Nos demais blocos todos os resultados foram satisfatórios. Mesmo quando havia choques de alocação, como nos blocos 3Q, 5O-A, 5R-A e 5R-B, eram baixos permitindo assim que o administrador de espaço físico remanejasse as turmas com choques. Nota-se que 50% dos blocos obtiveram soluções viáveis em todas as 20 execuções. Além disso, os desvios-padrão

da coluna ‘Final’ apresentados decorrentes das 20 execuções são relativamente baixos, o que indica robustez e confiabilidade da ferramenta proposta.

Somando os tempos de processamento de todos os blocos ao tempo necessário para a montagem de seções, a análise de resultados e a realocação de salas pós-processamento, estima-se que o tempo total para a execução do processo de alocação de salas no segundo semestre de 2016 consumiu cerca de 160 horas. Comparando com as 640 horas de trabalho necessárias para a realização do processo manualmente, verifica-se uma redução de 75% das horas necessárias. Vale ressaltar ainda que, foi a primeira vez em que o sistema de automatização proposto foi utilizado e foi necessário um período de adaptação ao novo processo, portanto espera-se que essa redução seja ainda maior nos próximos semestres.

Referências

- [1] BURKE, E. K. E VARLEY, D. B., *Space Allocation: an analysis of higher education requirements*, Lecture Notes in Computer Science, vol. 1408 (1997), pp. 20-36.
- [2] SCHAEFER, A., *A survey of automated timetabling*, Artificial Intelligence Review, vol. 13 (1999), pp. 87-127.
- [3] BARDADYN, V. A., *Computer-Aided School and University Timetabling: The New Wave*, Lecture Notes in Computer Science, vol. 1153 (1996), pp. 22-45.
- [4] CARTER, M. W. E TOVEY, C. A., *When is the Classroom Assignment Problem Hard?*, Operations Research, vol. 40 (1992), pp. 28-39.
- [5] DE WERRA, D., *An introduction to timetabling*, European Journal of Operational Research Society, vol. 19 (1985), pp. 151-162.
- [6] BURKE, E. K. E VARLEY, D. B., *Automating Space Allocation in Higher Education*, Lecture Notes in Artificial Intelligence, vol. 1585 (1998), pp. 66-73.
- [7] CZIBULA, O. GU, H. RUSSELL, A. E ZINDER, Y, *A Multi-Stage IP-Based Heuristic for Class Timetabling and Trainer Rostering*, Proceedings of the 10th International Conference on the Practice and Theory of Automated Timetabling, (2014), pp. 102-127.
- [8] MCCOLLUM, B., *A perspective on bridging the gap between theory and practice in university timetabling*, Lecture notes in computer science, vol. 3867 (2007), pp. 3-23.

- [9] CIRINO, R. B. Z, SANTOS, M. O, DELBEM, A. C. B., *Aplicação da Metaheurística AGC para o Problema de Alocação de Aulas à Salas*, Anais do XLVII SBPO (2015), pp. 1745-1755.
- [10] HE-NAN, Z. E SHAO-WEN, Z., *Solving UTP Containing Combining Classes using GA*, International Journal of u-and e-Service, Science and Technology, vol. 7 (2014), pp. 277-286.
- [11] LANDA SILVA, J. D., *Metaheuristic and Multiobjective Approaches for Space Allocation*, Thesis, School of Computer Science and Information Technology, University of Nottingham, Nottingham UK (2003).
- [12] QUEIROGA, E. V, JÚNIOR, T. L. B, CABRAL, L. A. F, DA COSTA, L. C. A, SUBRAMANIANA, A., *Problema de Alocação de Aulas: O caso da Central de Aulas da UFPB*, Anais do XLVII SBPO (2015), pp. 848-859.
- [13] SALES, E. S, MÜLLER, F. M, SIMONETTO, E. L., *Solução do Problema de Alocação de Salas Utilizando um Modelo Matemático Multi-índice*, Anais do XLVII SBPO (2015), pp. 2596-2607.
- [14] PAECHTER, B., *Mine's better than yours - comparing timetables and timetabling algorithms*, Proceedings of the 10th International Conference on the Practice and Theory of Automated Timetabling (2014), pp. 8-9.
- [15] SILVA, D. J. DA, SILVA, G. C. DA, *Heurísticas Baseadas no Algoritmo de Coloração de Grafos para o problema de alocação de salas em uma instituição de ensino superior*, Simpósio Brasileiro de Pesquisa Operacional, (2010), pp. 2839-2849.
- [16] GEN, M. E CHENG, R., *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, Inc. (2000).

- [17] LINDEN, R., *Algoritmos Genéticos*, Ciência Moderna, 3ª Edição (2012).
- [18] GOLDBERG, D. E., *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley Publishing Company, Inc. (1989).
- [19] MICHALEWICZ, Z., *Genetic algorithms + data structures = evolution programs*, Springer, 3rd Edition (1996).
- [20] ENGELBRECHT, A. P., *Computational Intelligence An Introduction*, John Wiley & Sons, Inc. (2007).
- [21] TOMASSINI, M., *A survey of Genetic Algorithms*, Annual Reviews of Computational Physics, vol. 3 (1995), pp. 3-4.
- [22] MAN, K. F. TANG, K. S. E KWONG, S., *Genetic Algorithms: Concepts and Design*, Springer (1999).

Publicações

As publicações resultantes deste trabalho são apresentadas a seguir.

Proposta de um Sistema Baseado em Computação Evolutiva para o Tratamento do Problema de Alocação de Espaço Físico: O Caso da Universidade Federal de Uberlândia

Guilherme Palhares Theodoro

Igor Santos Peretta

Keiji Yamanaka

Trabalho apresentado na XIV Conferência de Estudos em Engenharia Elétrica, Uberlândia, Brasil, 2016.

Utilização de Algoritmos Genéticos para o Problema de Alocação de Salas da Universidade Federal de Uberlândia

Guilherme Palhares Theodoro

Igor Santos Peretta

Keiji Yamanaka

Trabalho apresentado na XIII Encontro Nacional de Inteligência Artificial e Computacional, Recife, Brasil, 2016.

Disponível em: <http://www.lbd.dcc.ufmg.br/colecoes/eniac/2016/045.pdf>