

INSTITUTO FEDERAL

Goiás

Câmpus Anápolis

Gabriel Cotrim Ferreira

**Modelagem matemática e avaliação de um
algoritmo para um problema de designação
generalizada na construção de horários de aula**

Anápolis-GO

2025

Gabriel Cotrim Ferreira

**Modelagem matemática e avaliação de um algoritmo para
um problema de designação generalizada na construção de
horários de aula**

Trabalho de Conclusão de Curso apresentado
ao Instituto Federal de Goiás como parte
das exigências para obtenção do título de
Bacharel em Ciência da Computação..

Instituto Federal de Goiás - Câmpus Anápolis

Orientador: Prof. Dra. Kátia Cilene Costa Fernandes
Coorientador: Prof. Dr. Sérgio Daniel Carvalho Canuto

Anápolis-GO

2025



INSTITUTO FEDERAL
Goiás

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
SISTEMA INTEGRADO DE BIBLIOTECAS

**TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAÇÃO
NO REPOSITÓRIO DIGITAL DO IFG - ReDi IFG**

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia de Goiás, a disponibilizar gratuitamente o documento no Repositório Digital (ReDi IFG), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IFG.

Identificação da Produção Técnico-Científica

- | | |
|--|---|
| ☐ Tese | ☐ Artigo Científico |
| ☐ Dissertação | ☐ Capítulo de Livro |
| ☐ Monografia – Especialização | ☐ Livro |
| ☒ TCC - Graduação | ☐ Trabalho Apresentado em Evento |
| ☐ Produto Técnico e Educacional - Tipo: _____ | |

Nome Completo do Autor: Gabriel Cotrim Ferreira

Matrícula: 20201060140267

Título do Trabalho: Modelagem matemática e avaliação de um algoritmo para um problema de designação generalizada na construção de horários de aula

Autorização - Marque uma das opções

1. (X) Autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso aberto);
2. () Autorizo disponibilizar meu trabalho no Repositório Digital do IFG somente após a data ____/____/____ (Embargo);
3. () Não autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso restrito).

Ao indicar a opção **2 ou 3**, marque a justificativa:

- () O documento está sujeito a registro de patente.
() O documento pode vir a ser publicado como livro, capítulo de livro ou artigo.
() Outra justificativa: _____

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

- i. o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- ii. obteve autorização de quaisquer materiais incluídos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- iii. cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

_____, 11/02/2026.
Local Data

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
SISTEMA INTEGRADO DE BIBLIOTECAS

A handwritten signature in dark ink, reading "Gabriel Cotrim Ferreira". The signature is written in a cursive style with some capital letters.

Assinatura do Autor e/ou Detentor dos Direitos Autorais

Dados Internacionais de Catalogação na Publicação (CIP)

F383m FERREIRA, Gabriel Cotrim

Modelagem matemática e avaliação de um algoritmo para um problema de designação generalizada na construção de horários de aula. / Gabriel Cotrim Ferreira. – Anápolis: IFG, 2025.
37 f. il. color.

Orientadora: Prof. Dra. Kátia Cilene Costa Fernandes.
Coorientador: Prof. Dr. Sérgio Daniel Carvalho Canuto.

Trabalho de Conclusão de Curso – Instituto Federal de Educação, Ciência e Tecnologia de Goiás, Campus Anápolis – Bacharelado em Ciência da Computação, 2025.

1. Problema de designação generalizada. 2. Algoritmo genético. 3. Otimização combinatória. 4. Modelo matemático de designação.

I. FERNANDES, Kátia Cilene Costa (orient.).

II. CANUTO, Sérgio Daniel Carvalho (coorient.).

III. Título.

CDD 519.6

ATA DE DEFESA DE MONOGRAFIA

Na presente data realizou-se a sessão pública de defesa da Monografia intitulada **Modelagem matemática e avaliação de um algoritmo para um problema de designação generalizada na construção de horários de aula**, sob orientação de Katia Cilene Costa Fernandes, apresentada pelo aluno **Gabriel Cotrim Ferreira (20201060140267)** do Curso **Bacharelado em Ciência da Computação (Câmpus Anápolis)**. Os trabalhos foram iniciados às **18:05** do dia **04/09/2025** pela Professora presidente da banca examinadora, constituída pelos seguintes membros:

- **Katia Cilene Costa Fernandes** (Presidente)
- **Sergio Daniel Carvalho Canuto** (Examinador Interno)
- **Hugo Vinicius Leao e Silva** (Examinador Interno)

A banca examinadora, tendo terminado a apresentação do conteúdo da Monografia, passou à arguição do candidato. Em seguida, os examinadores reuniram-se para avaliação e deram o parecer final sobre o trabalho apresentado pelo aluno, tendo sido atribuído o seguinte resultado:

☒ Aprovado

☐ Reprovado

Nota : 6,8

Observação / Apreciações:

Proclamados os resultados pelo presidente da banca examinadora, foram encerrados os trabalhos e, para constar, eu **Katia Cilene Costa Fernandes** lavrei a presente ata que assino juntamente com os demais membros da banca examinadora.

Documento assinado eletronicamente por:

- **Katia Cilene Costa Fernandes** (534.077.331-00), em **05/09/2025 08:21:00** com chave **66aff0be8a4a11f0a215005056a537a4**.
- **Sergio Daniel Carvalho Canuto** (020.497.821-10), em **05/09/2025 08:25:27** com chave **05ab47368a4b11f0b49c005056a537a4**.
- **Hugo Vinicius Leao e Silva** (004.251.811-38), em **05/09/2025 09:18:01** com chave **5e1af4508a5211f0a215005056a537a4**.

Este documento foi emitido pelo SUAP. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse https://suap.ifg.edu.br/comum/autenticar_documento/ e informe os dados a seguir.

Tipo de Documento: Ata de Projeto Final

Data da Emissão: 05/09/2025

Código de Autenticação: 13bf4a



Lista de ilustrações

Figura 1 – Representação de um problema de designação.	15
Figura 2 – Grade-horária com as disciplinas alocadas gerada pelo algoritmo com os dados da Tabela 1.	27
Figura 3 – Função distribuição de quantidade de professores por dia.	29
Figura 4 – Função distribuição cumulativa da quantidade de dias trabalhados por cada professor.	30
Figura 5 – Quantidade de professores alocados em suas respectivas opções de prioridade de dias da semana.	31
Figura 6 – Quantidade de professores alocados na opção mais desejada disponível.	32
Figura 7 – Quantidade de professores alocados na opção menos desejada disponível.	32

Lista de tabelas

Tabela 1 – Dados de entrada - Representação de matérias e suas características . .	25
Tabela 2 – Tabela de prioridades	26

Lista de abreviaturas e siglas

AG	Algoritmo Genético
GAP	<i>Generalized Assignment Problem</i> (Problema de designação generalizada)
IFG	Instituto Federal de Educação, Ciência e Tecnologia de Goiás
SA	Simulated Annealing
TIC	Tecnologia de Informação e Comunicação

Resumo

O problema de designação é um problema clássico de otimização combinatória que tem o objetivo de minimizar o custo total ao atribuir n tarefas a m agentes, auxiliando na tomada de decisão. Neste trabalho, é apresentado um modelo de problema de designação generalizado associado à distribuição de horários de aula em uma instituição de ensino. Além disso, cria-se um algoritmo genético para resolver esse problema, acompanhado de uma análise dos resultados. Os resultados obtidos demonstram que o algoritmo é capaz de gerar grades horárias válidas e atender às restrições impostas, com tempo médio de execução consideravelmente pequeno. Essa abordagem proposta mostra-se eficaz para auxiliar na automatização do processo de geração de horários acadêmicos, podendo ser expandida para outros cenários de designação de horários.

Palavras-chave: Problema de designação generalizada, algoritmo genético, otimização combinatória, modelo matemático de designação.

Sumário

	Introdução	11
1	REFERENCIAL TEÓRICO	13
1.1	Problemas de Transporte e Designação	13
1.2	Algoritmo Genético	16
2	OBJETIVO	19
2.1	Objetivo Geral	19
2.2	Objetivo Específico	19
3	METODOLOGIA	20
4	MODELO MATEMÁTICO DE ALOCAÇÃO DE DISCIPLINAS EM UMA GRADE DE HORÁRIOS	22
5	CONSTRUÇÃO DE UM ALGORITMO GENÉTICO	24
6	ANÁLISE DOS RESULTADOS	29
7	CONSIDERAÇÕES FINAIS	34
	REFERÊNCIAS	36

Introdução

A otimização combinatória e inteira trata de problemas de maximização e minimização, ou seja, maximiza ou minimiza uma função objetivo envolvendo várias variáveis de decisão acompanhada de dois tipos de restrições, de igualdades/desigualdades e as restrições de integralidade sobre algumas ou todas as variáveis (FERNANDES, 2019).

O objetivo da otimização combinatória é encontrar um objeto ótimo a partir de um conjunto finito de objetos. Esses objetos candidatos são chamados de soluções viáveis, enquanto o ótimo é chamado de solução ótima (DU et al., 2022).

Dentre os problemas de otimização combinatória existe uma vertente voltada a problemas de alocação, denominado de Problema de Alocação/Designação Generalizada. Esse tipo de problema examina a atribuição de n tarefas para m agentes com o objetivo de minimizar o custo total de atribuição e maximizar a eficiência de forma que cada tarefa seja designada a exatamente um único agente, sendo sujeito a restrições de capacidade desses agentes (OLIVEIRA et al., 2019; CATTRYSSSE; WASSENHOVE, 1992).

É comum em situações onde as tarefas consomem diferentes quantidades de recursos dos agentes, e as soluções precisam respeitar restrições de capacidade, sendo aplicadas em contextos como logística, produção e agendamento. Por exemplo, o agendamento de horários de aula, que consiste em designar determinadas aulas para determinados horários e turmas (DIAS, 2018).

A literatura apresenta uma variedade de algoritmos, exatos ou heurísticas, com o intuito de encontrar soluções para esse tipo de problema. Quando as soluções ótimas são difíceis de achar, heurísticas são frequentemente usadas para produzir soluções próximas a soluções ótimas, denominadas de soluções subótimas, que tenham uma qualidade garantida (GROTSCHER; LOVÁSZ, 1995). Dentre essas heurísticas, há os algoritmos genéticos.

O algoritmo genético é uma técnica de otimização inspirada na teoria da evolução natural de Darwin (HOLLAND, 1992). Ele simula o processo de seleção natural, onde uma população de soluções (chamadas de indivíduos) evolui ao longo de gerações. As soluções são avaliadas por uma função de *fitness*, e as melhores são selecionadas para reprodução, passando por operações como cruzamento e mutação (MIRJALILI, 2019). O objetivo é explorar o espaço de soluções e, ao longo de várias iterações, encontrar soluções próximas ao ótimo, denominadas soluções subótimas, para problemas complexos.

O objetivo deste trabalho é propor um modelo matemático para o problema de designação

generalizada de horários de aula, visando otimizar a alocação de disciplinas e docentes, com prioridades previamente estabelecidas em uma grade horária. O modelo considera restrições como a carga semanal de cada disciplina, o número de turmas, a distribuição das disciplinas entre turmas e docentes, bem como a não sobreposição de aulas para professores e salas, garantindo a consistência e a viabilidade do cronograma acadêmico.

Para resolver esse problema, é desenvolvido um algoritmo genético que, com base nas necessidades do modelo matemático criado, traz uma solução subótima para o mesmo, ou seja, gera uma grade de horários com o intuito de otimizar as demandas solicitadas pelos docentes em relação aos dias da semana mais viáveis para atendê-los.

O referencial teórico é tratado no Capítulo 1, ele tem o objetivo de apresentar os conceitos e definições do problema de transporte e do problema de designação, bem como explicar o que é um algoritmo genético, como ele funciona e sua finalidade. O Capítulo 2 apresenta o objetivo geral e os objetivos específicos definidos para este trabalho. O Capítulo 3 foca em explicar o processo metodológico abordado, além de narrar a trajetória do desenvolvimento do trabalho.

O Capítulo 4 apresenta o modelo matemático criado para representar as características e necessidades do problema proposto, já no capítulo 5 contém um pseudo-código que representa o algoritmo criado a partir das demandas do modelo matemático definido no capítulo anterior.

Neste Capítulo 5 também traz os conceitos e implementações do algoritmo desenvolvido, além de comentar os formatos de entrada e saída dos dados. O Capítulo 6 apresenta os resultados obtidos, com gráficos para ajudar a visualizar a proficiência do algoritmo em cumprir os requisitos desejados. No Capítulo 7 estão as considerações finais do trabalho.

E, por último, são apresentadas as referências desse trabalho.

1 Referencial Teórico

A otimização combinatória e inteira envolve problemas que buscam maximizar ou minimizar uma certa função, denominada de função objetivo, com várias variáveis de decisão, onde o domínio é tipicamente finito, sujeito a duas categorias de restrições: as de igualdade ou desigualdade, e as de integralidade, que podem se aplicar a algumas ou a todas as variáveis (FERNANDES, 2019).

A otimização combinatória busca a melhor solução para o problema, denominada de solução ótima, dentre as diversas combinações possíveis existentes. Esse tipo de problema encontra várias aplicações do cotidiano, como nos problemas de transporte e de designação que são apresentados a seguir.

1.1 Problemas de Transporte e Designação

A programação linear é uma subárea da programação matemática, na qual as restrições dos problemas e a função objetivo são apresentadas por funções lineares. Assim, a formulação generalizada de problemas de programa linear é dada por:

$$\text{Maximizar } Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

sujeito a:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &\leq b_m \\ x_1, x_2, \cdots, x_n &\geq 0 \end{aligned}$$

Onde z é a função objetivo que representa uma medida de desempenho do sistema, n representa o número de variáveis de decisão e m o número de restrições funcionais. A partir de então pode-se dizer que um problema de programação linear é padrão se obtiver a maximização (ou minimização) da função objetivo, e ainda se as restrições vierem do tipo menor ou igual (ou, maior ou igual/igual). Ressalva-se ainda que as variáveis de decisão x_i assumem valores não negativos (LACHTERMACHER, 2016).

Uma classe especial de problemas de programação linear é o problema de transporte. Esse tipo de problema trata do envio de uma quantidade de produto de origens (por exemplo, fábricas) para destinos (por exemplo, depósitos). O objetivo é determinar a programação de uma rota que minimize o custo total e, ao mesmo tempo, satisfaça os limites de fornecimento e demanda.

Aplicações do problema de transporte podem ser estendidas a outras áreas de operações, entre elas controle de estoque, programação de empregos e designação de pessoal (TAHA, 2008). Assim o problema de transporte surge com a necessidade de definir a distribuição de um determinado produto o qual se encontra disponível em m origens, nas quantidades $a_i > 0$ com $i = 1, 2, \dots, m$, e é requerido em n destinos nas quantidades $b_j > 0$ com $j = 1, 2, \dots, n$ (MARINS, 2011; LACHTERMACHER, 2016).

A formulação generalizada do problema de transporte é dada por:

$$\text{Minimizar } \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (1.1)$$

sujeito a:

$$\sum_j^n x_{ij} \leq a_i, \quad i = 1, 2, \dots, m \quad (\text{balanço do produto nas origens}) \quad (1.2)$$

$$\sum_i^m x_{ij} = b_j, \quad j = 1, 2, \dots, n \quad (\text{balanço do produto nos destinos}) \quad (1.3)$$

$$x_{ij} \geq 0, \quad \text{com } i = 1, 2, \dots, m, \text{ e } j = 1, 2, \dots, n \quad (1.4)$$

onde c_{ij} é o custo do transporte de uma unidade da origem i para o destino j , a_i é a quantidade disponível na origem i . O b_j é a quantidade (recurso) requerida no destino j .

A função objetivo (1.1) minimiza o custo total para transportar um determinado produto da origem i para o destino j . As restrições (1.2) asseguram que a capacidade ofertada de cada origem i não ultrapasse o seu limite e as restrições (1.3) garantem que cada demanda seja atendida. As restrições (1.4) representam a não-negatividade das variáveis de atribuição x_{ij} , ou seja, x_{ij} é a quantidade do produto a ser transportado da origem i para o destino j .

Dentre os problemas de otimização combinatória onde todas as variáveis de decisão são binárias, ou seja, assumem valores 0 ou 1, há os problemas de designação ou de atribuição generalizada (*generalized assignment problem* - GAP). Esse tipo de problema muitas vezes é apresentado como um caso particular de problema de transporte (ÖNCAN, 2007; OLIVEIRA et al., 2019).

O Problema de Alocação Generalizada é um problema de otimização combinatória. Esse problema é uma generalização do problema de atribuição em que examina o custo mínimo de atribuição de n trabalhos para m agentes, onde cada trabalho é designado a apenas um agente, sujeito às restrições dos agentes.

Para melhor compreensão, será apresentado o modelo matemático generalizado desse tipo de problema.

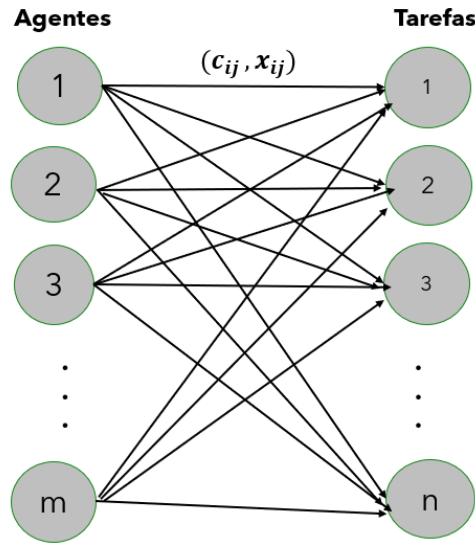


Figura 1 – Representação de um problema de designação.

Sejam I o conjunto formado pelos agentes que será designado para realizar determinadas tarefas que pertencem a um determinado conjunto J . Ou seja, $I = \{1, 2, 3, \dots, i, \dots, m\}$ é o conjunto de m agentes e $J = \{1, 2, 3, \dots, j, \dots, n\}$ é o conjunto de n tarefas (Figura 1).

A formulação generalizada do problema GAP é dada por:

$$\text{Minimizar } \sum_i^m \sum_j^n c_{ij} x_{ij} \quad (1.5)$$

sujeito a:

$$\sum_j^n a_{ij} \leq b_i, \quad i \in I \quad (1.6)$$

$$\sum_i^m x_{ij} = 1, \quad j \in J \quad (1.7)$$

$$x_{ij} = 0 \text{ ou } 1, \quad i \in I, \quad j \in J \quad (1.8)$$

$ondec_{ij}$ é o custo de atribuição da tarefa j ao agente i , a_{ij} a absorção de capacidade quando a tarefa j é atribuída ao agente i , ou seja, cada agente i gasta a_{ij} de um dado recurso (e.g., tempo) para executar a tarefa j . O b_j é a capacidade (recurso) disponível do agente i .

A função objetivo (1.5) minimiza o custo total das tarefas designadas aos agentes. As restrições (1.6) asseguram que a capacidade de cada agente não ultrapasse o seu limite e as restrições (1.7) garantem que cada tarefa será executada exatamente uma vez. As variáveis (1.8) de atribuição x_{ij} são iguais a um se o agente i executar a tarefa j , e zero, caso contrário.

Na literatura existem várias pesquisas com o objetivo de encontrar algoritmos de enumeração eficientes para achar uma solução ótima para problemas de designação. O trabalho de [Fraccaroli \(2018\)](#) apresenta um problema que designa tarefas que cada robô de monitoramento de plantações realizará e também coordenará todos os robôs durante a execução do sistema como um todo, otimizando o processo de coleta de dados.

Outro é [Masoud et al. \(2019\)](#) que modela e cria um algoritmo para resolver um problema de designação generalizada de carregamento como uma atribuição entre e-scooters e carregadores, ou seja, aloca as e-scooters aos carregadores com ênfase em minimizar a distância média percorrida dos carregadores para coletar as e-scooters. Quatro anos depois [Masoud et al. \(2023\)](#) desenvolvem um modelo matemático para o problema de alocação de e-scooters e carregadores que utiliza a metaheurística de Simulated Annealing (SA) para encontrar uma solução subótima. SA faz buscas pelo espaço de soluções e evita ficar preso em um ponto ótimo local ([MIRSADEGHI; KHODAYIFAR, 2021](#)).

[Silva \(2020\)](#) traz situações-problemas com demandas reais, em que analisa, a partir do desenvolvimento de um protótipo, o desempenho quanto ao uso de sistemas de recomendação na etapa de designação de profissionais para atendimento a serviços de Tecnologia da Informação e Comunicação - TIC, não apresentando um modelo matemático que represente essa situação.

Outros trabalhos usam problemas de designação para modelar o problema e que estão mais próximos do tema abordado, como [Dias \(2018\)](#) que aborda o problema de alocação de turmas a salas de aula, levando em conta a capacidade máxima de cada sala, além da distância entre prédios e departamentos. Já [ANDRADE, SCARPIN e STEINER \(2012\)](#) apresentam resultados de aplicação de modelos para geração de grades do curso de Engenharia de Produção da Universidade Federal do Paraná, na qual se observam as demandas do curso, visando respeitar as variáveis envolvidas no processo de elaboração da grade-horária.

1.2 Algoritmo Genético

Algoritmo Genético (AG) é um dos primeiros algoritmos estocásticos baseados em população propostos na história. O AG foi baseado na teoria da evolução de Charles Darwin, em que a sobrevivência das criaturas mais aptas e os seus genes foram simulados ([HOLLAND, 1992](#)). Como o AG é um algoritmo baseado em população, cada solução representa um cromossomo e cada parâmetro representa um gene ([MIRJALILI, 2019](#)).

Os principais operadores do AG são seleção, *crossover* e mutação. Juntos constituem os operadores genéticos que serão aplicados numa população inicial aleatória. O cromossomo precisa ter um formato que guarde informações relacionadas à solução que ele representa, sendo a codificação amplamente utilizada, a de *string* binária ([LAMBORA; GUPTA;](#)

(CHOPRA, 2019).

Assim, o AG começa com uma população inicial que inclui múltiplas soluções, representando os cromossomos dos indivíduos. Cada cromossomo tem um conjunto de variáveis, que simulam os genes. O principal objetivo da inicialização é espalhar as soluções pelo espaço de busca o mais uniformemente possível para aumentar a diversidade da população e melhorar as chances de encontrar regiões promissoras (MIRJALILI, 2019). A técnica de seleção é inspirada principalmente no conceito de seleção natural, com essa simples ideia o AG implementa a técnica da roleta que define probabilidade aos indivíduos e seleciona-os para a próxima geração com base em sua fitness (aptidão). Vale ressaltar que o método da roleta é apenas um dos vários operadores de seleção existentes na literatura. Este mecanismo simula a seleção natural do indivíduo mais apto na natureza (MIRJALILI, 2019).

Depois de selecionar os indivíduos usando um operador de seleção, eles precisam ser usados para criar a próxima geração. Na natureza, os cromossomos de indivíduos do sexo masculino e feminino são combinados para gerar um novo cromossomo. Isto é simulado combinando duas soluções selecionadas pelo método da roleta para produzir duas novas soluções no AG (MIRJALILI, 2019). Na literatura, existem diferentes técnicas para o operador de crossover, como a single-point e a double-point (SRINIVAS; PATNAIK, 1994). A mutação é o último operador evolucionário, no qual um ou vários genes são alterados após a criação das soluções-filho. O operador de mutação mantém a diversidade da população introduzindo uma nova camada de randomicidade. Este operador previne soluções de se tornarem similares e aumenta a probabilidade de evitarem soluções locais no AG (MIRJALILI, 2019).

Algoritmos genéticos podem ser usados para resolver problemas de alocação dos mais variados tipos. Bazzazi, Safaei e Javadian (2009) fazem uso de AG para resolver o problema de alocação de espaço de armazenamento em um terminal de *containers*. O armazenamento temporário de *containers* é um dos serviços mais importantes em um terminal de *containers* pois uma rápida armazenagem e retirada dos mesmos é essencial para o desempenho econômico.

Alguns fatores levantados em (BAZZAZI; SAFAEI; JAVADIAN, 2009) são que para um *container* ser alocado ou retirado pode ser necessário mover um ou mais *containers* para liberar o acesso, tal processo também inclui o tempo de coleta, movimentação e deposição do *container*. O modelo proposto pelo artigo possui inúmeras restrições de igualdade e a implementação de um AG é proposta para alcançar as soluções viáveis, além de ser definida como uma das vantagens marcantes do mesmo. Para determinadas amostras, resolveram o problema com auxílio de algoritmos exatos para verificar a eficiência dessas soluções geradas pelo AG proposto e constataram uma diferença de aproximadamente 5%

em relação às soluções ótimas.

Outro trabalho que utiliza um algoritmo genético para resolver um problema de alocação de salas é (CHEN et al., 2021). Nele é apresentado um algoritmo genético para a geração de um horário, o sistema desenvolvido segue o fluxo padrão desse tipo de algoritmo, selecionando, cruzando e aplicando mutação nos indivíduos até gerar um resultado eficiente ou atingir o limite de gerações. Além disso, ao final deste processo, ainda cabe a opção de fazer ajustes manuais no horário, caso seja necessário.

É interessante citar a estrutura do software desenvolvido por Chen et al. (2021), que possui um cliente, uma camada de aplicação, uma camada intermediária e o sistema operacional. Isso permite que o usuário tenha apenas acesso superficial ao software, enquanto todos os processos principais são executados fora de seu alcance na camada de aplicação. Informações como classes, salas, turmas e professores ficam armazenadas em um banco de dados na camada intermediária, é através delas que o AG monta os indivíduos que serão usados para a criação de um horário.

2 Objetivo

2.1 Objetivo Geral

Este trabalho tem como objetivo modelar um problema generalizado de atribuição para a distribuição de horários de aula em uma instituição de ensino, bem como criar um algoritmo genético que gere soluções subótimas.

2.2 Objetivo Específico

Para melhor compreensão dos objetivos desta pesquisa, é apresentada a seguir as suas especificidades.

1. Realizar uma pesquisa bibliográfica sobre problemas generalizados de atribuição, bem como algoritmos genéticos.
2. Criar um modelo matemático de designação generalizada para representar um problema de distribuição de horários de aulas em cursos do IFG - Câmpus Anápolis.
3. Entender as restrições e as necessidades no que se refere à construção de horários de aula e atender, se possível, às solicitações do tomador de decisão que estarão nas restrições do modelo criado.
4. Compreender o processo de coleta de dados de entrada para iniciar a construção de horários de aula dessa instituição.
5. Criar um algoritmo genético para gerar soluções para o modelo.
6. Aprimorar o algoritmo adicionando recursos conforme julgado necessário, para se adequar mais ao modelo ou melhorar os resultados.

3 Metodologia

Primeiramente, a pesquisa concentra-se na compreensão dos conceitos fundamentais de otimização, utilizando como base obras de referência, como as de (ARENALES et al., 2007) e (LACHTERMACHER, 2016). Em seguida, realiza-se uma revisão bibliográfica direcionada aos Problemas Generalizados de Atribuição, com ênfase especial no problema de alocação de turmas, conforme discutido por (DIAS, 2018).

Posteriormente, a investigação avança para o estudo do tratamento dos Problemas Generalizados de Atribuição no contexto de *timetabling*, o que possibilitou o desenvolvimento de estratégias de modelagem matemática voltadas à construção de horários escolares. Esse processo exige uma dedicação significativa durante a pesquisa. Vale ressaltar que a modelagem elaborada considera as particularidades do Instituto Federal de Goiás (IFG) – Campus Anápolis, com base nas informações obtidas em entrevistas com o servidor responsável pela organização dos horários de aulas da instituição.

Em caráter metodológico exploratório, inicialmente são explorados os potenciais objetivos e restrições associados ao problema de geração de horários, que podem ser sumarizados pela seguinte questão de pesquisa: Quais são as necessidades observadas no problema de geração de horários? Para abordar essa questão, foi utilizado o Problema de Atribuição Generalizada (*Generalized Assignment Problem* - GAP) como base para a construção de um modelo matemático preliminar. Esse modelo inicial serve como ponto de partida para o entendimento do problema e o desenvolvimento do projeto?

Durante essa fase, são investigadas diversas técnicas, incluindo algoritmos bioinspirados, com o objetivo de explorar diferentes possibilidades para otimizar a alocação das disciplinas nos horários propostos.

Para resolver o problema apresentado, são utilizadas técnicas baseadas em algoritmos genéticos para a geração de horários de aula. No contexto deste estudo, cada nova tabela de horários gerada é avaliada de acordo com o método de avaliação estabelecido, sendo classificadas como superiores, inferiores ou equivalentes às soluções anteriores. Isso permite uma exploração contínua e iterativa das possíveis soluções para a otimização da alocação de horários.

No presente trabalho, a técnica principal adotada é o método da roleta, em que cada indivíduo tem uma probabilidade de ser selecionado com base em sua avaliação. Assim, a roleta é formada de maneira que os indivíduos com melhores classificações ocupam uma maior proporção, aumentando suas chances de serem escolhidos para a próxima geração. Esse processo garante uma evolução contínua das soluções, mantendo um equilíbrio entre

a exploração de novas possibilidades e a preservação das melhores soluções já encontradas. O operador de crossover adotado consiste em uma adaptação do crossover ponto a ponto aplicada a um problema de permutação. Diferentemente do método clássico, que utiliza um ponto de corte fixo no cromossomo, seleciona-se aleatoriamente uma disciplina, promovendo a troca das posições correspondentes a essa disciplina entre duas grades horárias, consideradas como indivíduos. Caso, após a permutação, algum dos horários atribuídos à disciplina selecionada resulte em conflito, isto é, seja alocado em um horário já ocupado na grade, uma nova posição válida é escolhida de forma aleatória, assegurando a viabilidade da solução e o respeito às restrições do problema.

Uma grade horária genérica, sem a atribuição inicial de disciplinas, é considerada um dos dados de entrada fundamentais no processo de modelagem. Essa grade contém a estrutura básica dos horários semanais de aula, sem definir quais disciplinas ocuparão cada período. O papel do algoritmo é realizar a alocação de maneira otimizada, buscando distribuir as disciplinas nos horários disponíveis de acordo com as necessidades e restrições impostas pelo problema. Este método de inicialização aleatória da população garante uma maior cobertura do espaço de busca, reduzindo vieses iniciais e favorecendo a exploração global da solução.

Para o ser humano, a impossibilidade de alocar mais de uma aula no mesmo horário para a mesma turma é algo que é automaticamente compreendido como inviável, mas que o algoritmo precisa atender ao modelo criado, por meio de restrições explícitas no código. Além disso, outras restrições importantes foram inseridas, como a necessidade de respeitar a carga horária de cada disciplina, evitar a alocação de professores em dois lugares simultaneamente ou que dois professores sejam alocados em um mesmo lugar ao mesmo tempo. Cada uma dessas limitações exige uma descrição formal precisa, para que o algoritmo seja capaz de gerar soluções que não só maximizem a eficiência da alocação, mas também respeitem as condições práticas e operacionais do contexto escolar.

Cada indivíduo, que representa uma grade horária, é modelado como uma matriz de dimensão $5 \times 7 \times 9 \times 2$, refletindo a estrutura multidimensional do problema. Os experimentos foram conduzidos com uma população inicial de 200 indivíduos, ao longo de 100 gerações, adotando-se uma taxa de mutação de 2%. Em cada geração, os dois melhores indivíduos, de acordo com a função de aptidão, foram preservados por meio da estratégia de elitismo. O algoritmo genético desenvolvido apresenta desempenho computacional satisfatório, com tempo médio de execução de aproximadamente 1 min 24 s (um minuto e vinte e quatro segundos), sendo capaz de retornar uma grade horária válida ao final do processo evolutivo.

Esses ajustes e formalizações são essenciais para garantir que o modelo seja capaz de lidar com a complexidade do problema e gerar soluções viáveis e aplicáveis na prática.

4 Modelo Matemático de Alocação de Disciplinas em uma Grade de Horários

Nesta seção, apresenta-se um modelo matemático de problema de designação generalizada de horários de aula, considerando que existem mais de uma turma e que professores podem ministrar mais de uma disciplina. Assim, o modelo matemático, denominado por (P) , é criado com o intuito de otimizar os horários, obedecendo a restrições básicas de um problema de designação, bem como restrições exigidas pelo tomador de decisão.

Dados os conjuntos I formado pelos horários dos dias da semana, J , formado pelos dias da semana que são considerados dias letivos, K , o conjunto das disciplinas a serem alocadas, L tem como elementos as turmas e, por último, D formado pelos docentes que ministram as disciplinas.

A formulação do modelo matemático genérico que representa o nosso estudo com intuito de alocar as disciplinas numa grade de horários da melhor maneira possível tem como

função objetivo $Z = \sum_i^I \sum_j^J \sum_k^K \sum_l^L \sum_d^D C_{ijkl} X_{ijkl}$, onde C_{ijkl} representa o grau de importância de uma disciplina k , de um docente d , ser alocado em um determinado horário i , do dia da semana j de uma turma l .

Considera-se que quanto maior o valor de C_{ijkl} melhor é a alocação da disciplina naquele horário e dia da semana. Além disso, define-se X_{ijkl} , a variável de decisão binária do modelo, ou seja, $X_{ijkl} \in \{0, 1\}$ que indica se a disciplina k da turma l de um certo professor d está ou não alocada no horário i do dia da semana j .

Assim, segue o modelo (P) de designação de disciplinas para uma determinada grade horária, maximizando o grau de importância total das disciplinas serem alocadas.

$$(P) \quad \text{Maximizar } Z = \sum_i^I \sum_j^J \sum_k^K \sum_l^L \sum_d^D C_{ijkl} X_{ijkl} \quad (4.1)$$

sujeito a:

$$\sum_i^I \sum_j^J X_{ijkl} = A_k, \forall k \in K \text{ e } \forall l \in L \quad (4.2)$$

$$\sum_k^K X_{ijkl} \leq 1, \forall i \in I, \forall j \in J \text{ e } \forall l \in L \quad (4.3)$$

$$X_{ijkl} + X_{(i+1)jkl} \leq 1, \text{ se } b_{kl} = 0, \forall i \in I \text{ e } (i+1) \leq i_{max}, \quad (4.4)$$

$\forall j \in J, \forall k \in K \text{ e } \forall l \in L$

$$X_{ijkl} + X_{(i+a)jkl} \leq 1, \text{ se } b_{kl} = 1, \forall a \in \mathbf{Z}, a \geq 2 \text{ e } (i+a) \leq i_{max}, \quad (4.5)$$

$\forall i \in I, \forall j \in J, \forall k \in K \text{ e } \forall l \in L$

$$\sum_k^K \sum_l^L X_{ijkl} \leq 1, \forall i \in I, \forall j \in J \text{ e } \forall d \in D \quad (4.6)$$

$$X_{ijkl} \in \{0, 1\}, \forall i \in I, \forall j \in J, \forall k \in K, \forall l \in L \text{ e } \forall d \in D \quad (4.7)$$

Onde :

i - representa um determinado horário de um dia da semana, ou seja, $i \in I$.

j - representa o dia da semana, ou seja, $j \in J$.

k - representa uma determinada disciplina de uma turma, $k \in K$.

l - representa uma determinada turma, ou seja, $l \in L$.

d - representa um determinado docente que ministra uma determinada disciplina, ou seja, $d \in D$.

i_{max} - representa o último horário de um dia da semana, ou seja, o maior valor de $i \in I$.

A_k - representa a quantidade total de aulas semanais de uma disciplina k .

$b_{kld} \in \{0, 1\}$ - representa se determinada disciplina k de uma turma l de um professor d terá horários de aula geminados (bloqueados) ou não.

Quanto às restrições do problema (P) , a restrição (4.2) garante que a quantidade total de aulas semanais (A_k) de uma determinada disciplina k foi atendida. Já a (4.3) garante que em um determinado horário i do dia da semana j seja alocado por, no máximo, uma disciplina k . A restrição (4.4) garante que as aulas da disciplina k não sejam geminadas (bloqueadas) e que não ocorram duas aulas dessa mesma disciplina k no mesmo dia j . Caso contrário, a restrição (4.5) garante que a disciplina k deve ter aulas geminadas (bloqueadas). Temos também a restrição (4.6) que garante que um determinado professor p em um determinado horário i de um dia j não esteja em mais de uma turma, ou seja, em dois lugares ao mesmo tempo e a (4.7) garante que a variável só pode assumir um dos valores, zero ou um.

5

Construção de um Algoritmo Genético

A partir da construção do modelo matemático correspondente ao problema de designação (P), inicia-se o estudo para o desenvolvimento de um algoritmo genético voltado para a resolução do referido problema.

O algoritmo, inicialmente, gera uma população de soluções preliminares, que consiste em uma série de horários avaliados com base em critérios previamente definidos. A seguir, são formados pares de horários, entre os quais ocorre a permutação de elementos selecionados, com o intuito de explorar novas soluções.

Esse processo iterativo se repete por um número predefinido de gerações, com o objetivo de refinar as soluções e selecionar as mais bem avaliadas. Ao final, o algoritmo converge para um conjunto de soluções subótimas, adequadas às restrições e aos parâmetros do problema.

Como se trata de um algoritmo genético que inerentemente busca o melhor resultado, ou seja, maximizar a nota (grau de importância) dos horários gerados, no final de cada iteração, novos indivíduos são criados para a próxima geração utilizando técnicas de seleção e de cruzamento, dando prioridade aos que possuem uma classificação mais alta. Técnicas de mutação são úteis para evitar que o algoritmo fique preso em algum ponto máximo local, tendo em vista que se trata de um problema de maximização.

Segue o pseudo-código do algoritmo desenvolvido para achar uma solução subótima na geração de uma grade de horário do problema (P).

-
1. $B = \text{iniciaPopulação}(\text{valores}, \text{tamanho})$
 2. **Para** $\text{cont} \leftarrow 1$ até quantCiclos
 3. $B^* \leftarrow \text{ordena por nota}(B)$
 4. $A \leftarrow \text{elite}(B^*, y)$
 5. $\text{pares} \leftarrow \text{roleta}(B^*, A)$
 6. $\text{filhos} \leftarrow \text{cruzamento}(\text{pares})$
 7. $B^* \leftarrow \text{mutacao}(\text{filhos}, \text{canhe de mutacao})$
 8. $B^* \text{ concatena}(\text{elite})$
 9. **Fim Para**
 10. $\text{gravaResultado}(\text{melhor}(B^*))$
-

Na Linha 1 o algoritmo gera aleatoriamente uma população inicial de horários de aula, o tamanho da população é definido previamente. O código tem um looping iterativo da Linha 2 até a 9 que será executado um número predeterminado de vezes.

A Linha 3 ordena as grades em ordem decrescente com base na forma de pontuação utilizada, na Linha 4 a variável A recebe os y melhores resultados para serem a elite que vai ser conservada para a próxima iteração. As Linhas 5 e 6 são responsáveis por gerar aleatoriamente os pares e então cruzar eles, respectivamente. Então, na Linha 7 executa a mutação, que possui uma probabilidade predefinida de acontecer. A elite então é concatenada à lista de novos indivíduos para o próximo *looping* na linha 8. Já na Linha 10 ocorre a gravação do melhor resultado obtido em um arquivo, após a execução do código.

Portanto, o algoritmo desenvolvido é capaz de gerar grades de horários válidos baseando-se nas restrições definidas, e utilizando como entrada uma lista das matérias que irão compor a mesma.

Além de ser capaz de dar uma nota (grau de importância) para esses horários, a forma de ponderação é baseada no modelo matemático (P). O programa é capaz, também, de lidar com turnos (matutino, vespertino, integral, noturno).

A linguagem de programação escolhida para esse trabalho de pesquisa foi a GNU Octave, que foi desenvolvida para a computação matemática. Possui uma interface em linha de comando para a solução de problemas numéricos, lineares e não lineares, também é usada para problemas numéricos. Essa linguagem foi escolhida devido à sua facilidade de descrição e resolução de problemas vetoriais.

O software desenvolvido recebe como entrada uma tabela contendo a lista das matérias que serão geradas no horário, bem como o número de aulas semanais, o turno, o professor que ministra essa aula, a que turma ela pertence e se as aulas serão bloqueadas ou não, como mostra a Tabela 1. Essas informações são utilizadas para criar uma relação entre as disciplinas, evitando que alguma restrição seja descumprida. Assim, sendo capaz de gerar uma grade de horários.

Tabela 1 – Dados de entrada - Representação de matérias e suas características

Matéria	Carga Horária	Turma	Turno	Professor	Aula Blocada
Int. Ciênc. Comp.	1	1	Matutino	Hugo	1
Mat.Comp.	2	1	Matutino	Hugo	0
Pesq. Oper.	3	1	Matutino	Loja	0
Inf. Bás.	1	1	Matutino	Hugo	1
Informática	2	1	Matutino	Alessandro	1
Prog. Web	3	2	Matutino	Alessandro	1
Arquitetura I	2	2	Matutino	Hugo	1
Prog. Orien. Obj.	3	2	Matutino	Alexandre	0
Cálculo I	3	2	Matutino	Maria	0

O software também recebe como entrada uma tabela de pesos que representa o nível de interesse de cada professor em dar ou não aula em cada dia da semana, como pode

ser visto na Tabela 2. Assim, quanto maior o peso, melhor é a preferência do docente por aquele dia. Nos testes, a tabela foi feita de forma arbitrária, mas seu objetivo é concentrar a maior quantidade de aulas de cada professor nos dias que ele achar mais propícios e, se possível, não trabalhar nos dias com pesos menores. Através dos testes, percebeu-se que a disposição de prioridades dos dias de um professor pode influenciar na quantidade de dias que ele irá trabalhar na semana. Se um dia de prioridade alta for muito requisitado, as aulas ministradas podem ser alocadas em outros dias, resultando em uma quantidade maior de dias em que o professor irá trabalhar. O contrário também pode acontecer, se um professor der preferência a dias pouco concorridos, há mais chance de trabalhar menos dias na semana.

Tabela 2 – Tabela de prioridades

	Segunda	Terça	Quarta	Quinta	Sexta
Alessandro	10	9	8	2	1
Alexandre	8	10	9	2	1
Hugo	9	2	10	8	1
Loja	2	8	9	10	1
Maria	1	2	8	9	10

Pela Figura 2 observa que essa grade de horários gerada pelo algoritmo com base nos dados de entrada apresentados na Tabela 1 e na Tabela 2 pode-se observar que várias características provenientes dos dados contidos nela e que são usadas como restrição, como o fato de existir duas turmas - a Turma 1 e a Turma 2 - implicar na existência de dois blocos, o primeiro referente à Turma 1 e o segundo referente à Turma 2, com cada matéria estando no horário da sua respectiva turma. Também, nota-se que matérias de diferentes turmas mas que possuem o mesmo professor não foram alocadas no mesmo horário, como pode ser observado, por exemplo, nas matérias Informática e Programação Web, que são de turmas diferentes mas ministradas pelo mesmo professor.

SEGUNDA	TERÇA	QUARTA	QUINTA	SEXTA
---------	-------	--------	--------	-------

Turma 1

	Informática	Pesquisa Operacional.	Matemática Computacional	
	Informática	Informática Básica	Pesquisa Operacional	
	Introdução a Ciência da Computação	Matemática Computacional	Pesquisa Operacional	

Turma 2

Programação Web	Arquitetura 1	Programação Orientada a Objeto		
Programação Web	Arquitetura 1	Calculo 1		Calculo 1
Programação Web	Programação Orientada a Objeto	Programação Orientada a Objeto		Calculo 1

Figura 2 – Grade-horária com as disciplinas alocadas gerada pelo algoritmo com os dados da Tabela 1.

Em destaque também que as matérias tendem a estar juntas, isso se deve à forma de ponderação abordada pelo software que dá notas maiores aos resultados que possuem essa característica. O inverso também pode ocorrer, fazendo as aulas da mesma matéria ficarem distantes uma da outra. O uso da variável de bloqueio representa a intenção de determinada matéria ser, ou não, dada em aulas seguidas. Pois, segundo alguns professores, há matérias em que há uma preferência em evitar que as aulas fiquem próximas umas das outras.

Os primeiros testes são realizados em uma escala mais reduzida, com o foco exclusivamente na geração de uma grade horária para uma única turma, sem levar em consideração professores, outras turmas ou turnos, visando apenas o cumprimento dos requisitos mínimos. A quantidade de aulas de cada disciplina deve corresponder à sua carga horária, e cada horário de aula deve ter, no máximo, uma disciplina alocada.

Com essas duas restrições e a primeira ideia de problema desenvolvida nesta pesquisa implementadas no software, obtém-se o primeiro vislumbre de como os horários seriam gerados. Além disso, métodos de ponderação foram acrescentados no algoritmo para atender, por exemplo, se as disciplinas seriam ou não bloqueadas. À medida que a pesquisa avança, o software é progressivamente aprimorado, aproximando-se cada vez mais de uma solução eficiente que atenda às demandas solicitadas pelo modelo.

Uma vez que, por se tratar de um algoritmo genético, o mesmo código com as mesmas entradas pode gerar soluções distintas para atender à mesma demanda e atingir o mesmo objetivo. Isso ocorre devido à característica desse tipo de algoritmo, cujo objetivo principal é alcançar a melhor pontuação possível, e não necessariamente uma solução única. Fatores como o número de gerações e a quantidade de indivíduos

presentes na população influenciam diretamente a eficiência na busca por melhores soluções, além de impactar o tempo de execução de cada iteração, o que, por sua vez, afeta a duração total do programa.

6

Análise dos Resultados

Para os testes foi utilizado um notebook da marca ACER, modelo Aspire VX5-591G-54PG com um processador Intel Core i5-7300HQ 2.5GHz, memória RAM de 8GB DDR4, SSD Kingston NV3 1TB M.2 2280 NVMe. O código do algoritmo genético foi desenvolvido em GNU Octave, uma linguagem computacional de código aberto desenvolvida para computação matemática.

O algoritmo foi executado por 50 instâncias com os mesmos dados de entrada. O propósito dessa quantidade de execuções foi avaliar o algoritmo genético criado. Segue alguns gráficos e análises dos mesmos que demonstram o desempenho do algoritmo.

A Figura 3 mostra, após a execução do algoritmo, a quantidade de professores designados para trabalhar em cada dia da semana. O objetivo desse gráfico é contribuir com a gestão para planejamento de encontros/reuniões que atendam um quantitativo de professores significativo que já estejam na instituição naquele dia. Pode-se observar que em 60% das instâncias os dias que mais concentraram professores são quinta e sexta, totalizando 5 (cinco) professores. O dia com maior concentração de professores ocorre na quarta-feira, com 8 (oito) professores simultâneos, porém isso ocorreu em apenas 4% das instâncias.

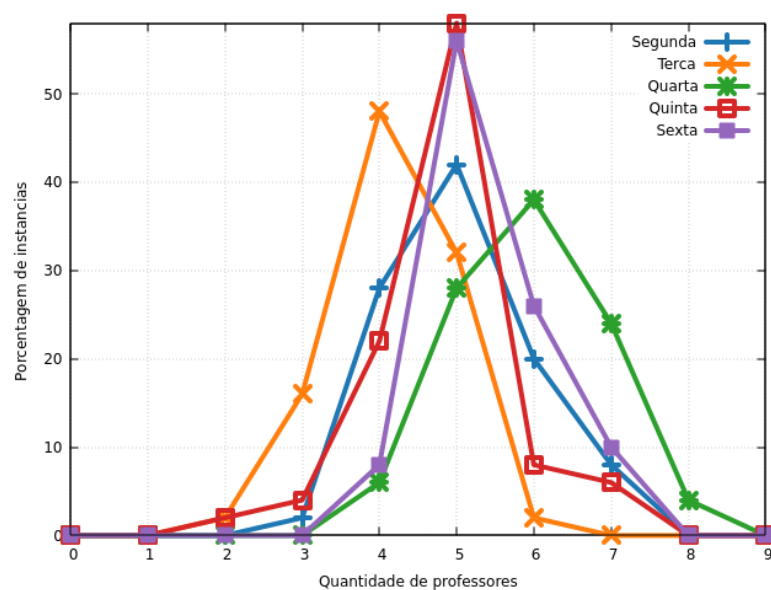


Figura 3 – Função distribuição de quantidade de professores por dia.

A Figura 4 representa a quantidade de dias da semana trabalhados por cada professor.

O gráfico demonstra a dispersão de dias que cada professor foi designado a trabalhar, entre os diferentes horários criados. Observa-se que a maioria dos professores, em até 100% das instâncias, trabalhou no máximo 4 dias por semana. Em até 50% dos casos, o professor 4 trabalhou em no máximo 3 dias da semana. O que se destacou de pior caso foi que o professor 4, em até 85% das instâncias, trabalha em até 4 dias da semana. Isso era esperado, pois ele é um dos professores que têm maior carga horária e quantidade de disciplinas. Já o professor 5, em nenhuma das instâncias, trabalhou mais do que dois dias da semana. Além disso, pelo menos 50% dos casos, 7 dos 9 professores trabalharam em até 3 dias da semana.

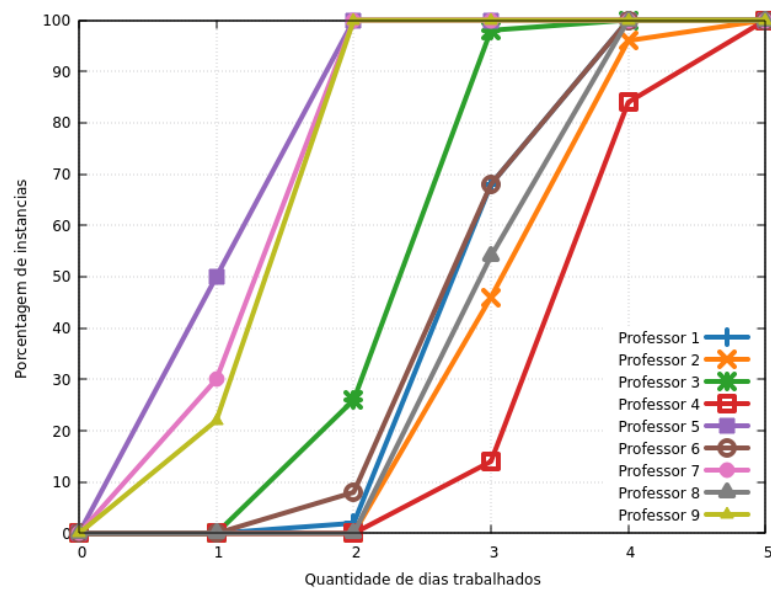


Figura 4 – Função distribuição cumulativa da quantidade de dias trabalhados por cada professor.

A Figura 5 representa as porcentagens de dias alocados aos professores, tendo como referência a preferência que eles tinham em relação a cada dia. Pode ser observado que, em até 95% das instâncias, no máximo três professores trabalharam nos dois dias de menor preferência para eles.

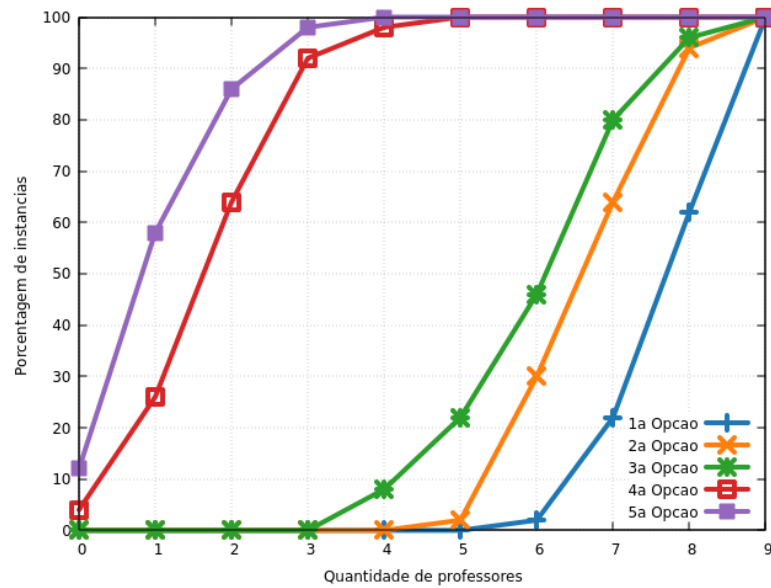


Figura 5 – Quantidade de professores alocados em suas respectivas opções de prioridade de dias da semana.

Note que, no mínimo, 6 professores trabalharam no seu dia de maior preferência e que, em 60% dos casos, 8 dos 9 professores do experimento trabalharam no dia que foi considerado sua principal escolha. Em mais de 90% dos casos, praticamente todos os professores trabalharam em seus dias de segunda e terceira maior preferência. Além disso, não houve menos do que 4 e 5 professores que trabalharam nesses dias, respectivamente.

O gráfico da Figura 6 é semelhante ao gráfico da Figura 5, só que em vez de mostrar todos os dias da semana, mostra apenas a alocação de maior prioridade feita para cada professor, desconsiderando os outros dias. Isso ajuda a ter uma noção da satisfação das necessidades dos professores. Note que a quarta e a quinta opção possuem 0 ocorrências em 100% das instâncias, isso significa que em todo o teste feito, mesmo no pior caso, todos os professores trabalharam em algum dos seus 3 dias de maior preferência. Observe que a linha que representa a primeira opção dos professores se encontra mais à direita no gráfico, onde se concentra a maior quantidade de professores satisfeitos. Note que a segunda e a terceira opção não tiveram mais do que dois professores trabalhando em cada um desses dias. Com isso, pode-se concluir que a grande maioria dos professores conseguiu trabalhar no dia que era considerado sua primeira opção.

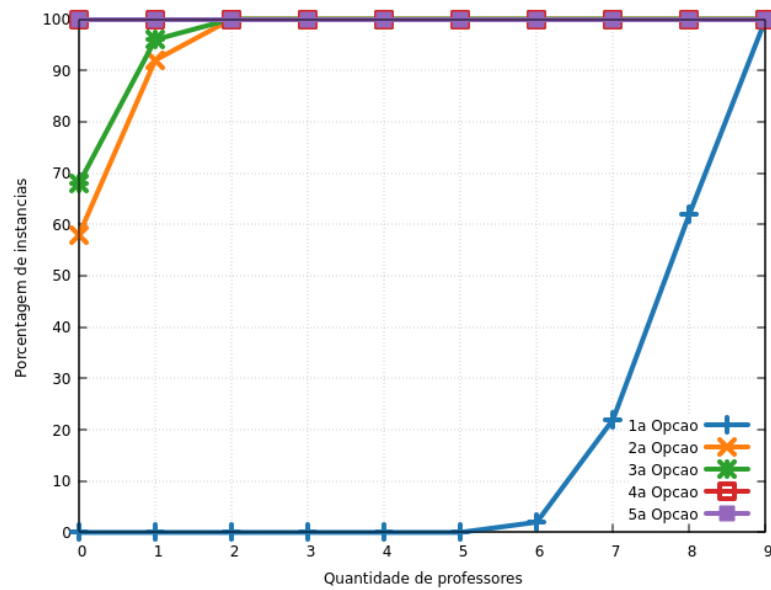


Figura 6 – Quantidade de professores alocados na opção mais desejada disponível.

O gráfico da Figura 7 contém apenas o dia menos desejado designado para cada professor trabalhar. Diferente da 6, que utiliza o limite superior dos dados recolhidos, este gráfico foca no limite inferior. A primeira coisa que pode ser notada é que as linhas do gráfico parecem estar mais espaçadamente distribuídas, se comparadas com a figura anterior.

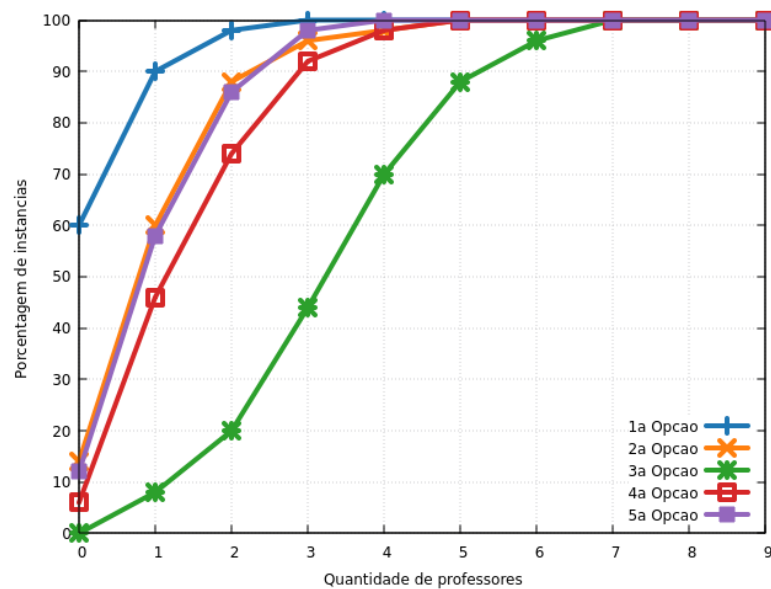


Figura 7 – Quantidade de professores alocados na opção menos desejada disponível.

Como esperado, há poucos resultados englobando a primeira opção, já que para isso ser possível o professor teria que trabalhar apenas um dia na semana, sendo esse o seu dia de maior preferência. A quarta e a quinta opção também apresentam uma concentração não muito grande de professores, o que indica que o algoritmo

prioriza alocar os professores nos dias da semana de maior prioridade para os mesmos, a segunda opção apresenta uma distribuição semelhante à da quinta opção. Em aproximadamente 95% dos casos, até 6 professores tiveram seu terceiro dia de aula de melhor escolha como sendo o dia de menor prioridade que tiveram que trabalhar.

7

Considerações Finais

O software cumpre com o seu papel pois, através de um arquivo de entrada que contém as informações necessárias, é gerada uma grade horária viável e subótima. Assim, a parte principal do trabalho foi cumprida.

Inicialmente o software implementava um método que favorecia apenas a alocação de aulas de uma mesma disciplina em blocos, porém, para algumas disciplinas, era desejável que as aulas não ocorressem de forma sequencial. Por esse motivo, foi implementada uma nova restrição, que tem como ponto principal as aulas não serem bloqueadas, ou seja, não ocorrer toda a carga horária semanal em horários consecutivos do mesmo dia da semana.

Os professores podem escolher se determinada disciplina que irão ministrar será bloqueada ou não, isso é representado no modelo matemático como uma restrição, porém, diferente das outras restrições, o seu descumprimento não resulta em um resultado inviável. Quando uma disciplina é marcada para não ter aulas bloqueadas, essa ocorrência será fortemente desincentivada, mas ainda poderá acontecer caso o melhor resultado encontrado não possa cumprir essa restrição para alguns docentes.

Outro ponto trabalhado foi a utilização de uma matriz de pesos personalizada para cada professor, com o intuito de atender às demandas da disponibilidade de dias da semana solicitadas pelos mesmos, assim atendendo à otimização da função objetivo do problema P criado e pela análise dos resultados percebe-se que, na maioria das instâncias, essa demanda foi atendida satisfatoriamente.

Como trabalhos futuros, recomenda-se ampliar os experimentos para instâncias de maior complexidade, incorporando restrições adicionais, como disponibilidade de laboratórios, prioridade para alunos concluintes, compatibilização de horários entre docentes e discentes, além de cenários com maior número de turmas, cursos e professores. A expansão permitirá avaliar a escalabilidade do algoritmo genético, incorporar novas restrições, identificar limitações de complexidade e aproximar-se de cenários reais em instituições de ensino. Também se propõe comparar sistematicamente a abordagem com técnicas como Simulated Annealing, Busca Tabu e métodos exatos de programação inteira, validando sua eficiência relativa quanto a tempo de execução, qualidade das soluções e robustez em diferentes configurações de dados.

Também se recomenda o desenvolvimento de uma interface gráfica intuitiva que permita à equipe administrativa interagir diretamente com o sistema, realizar ajustes

manuais e visualizar os impactos de diferentes parâmetros de entrada. Por fim, a integração do algoritmo a sistemas acadêmicos já utilizados pelas instituições poderia viabilizar sua adoção em larga escala, fortalecendo a relevância prática da pesquisa e abrindo espaço para investigações adicionais sobre usabilidade, aceitação por parte dos usuários e impacto organizacional da ferramenta.

Além disso, futuras extensões com o uso de Large Language Models (LLMs) podem apoiar a escolha dentre as múltiplas potenciais soluções mais adequadas dentre aquelas localizadas na fronteira de Pareto de um modelo multiobjetivo. A partir de um prompt em que o usuário explicita suas preferências (por exemplo, minimizar o número de dias trabalhados por cada professor, equilibrar a distribuição de aulas entre turmas ou priorizar determinadas restrições pedagógicas), o LLM pode interpretar essas diretrizes e indicar quais soluções melhor atendem aos critérios definidos. Assim, além de ordenar ou sugerir a alternativa mais alinhada às necessidades institucionais, o modelo também pode explicar de forma clara os compromissos assumidos entre objetivos conflitantes, tornando o processo de decisão mais transparente e acessível para gestores acadêmicos.

O código do algoritmo utilizado nesta pesquisa, dados de entrada, bem como os resultados gerados, estão disponíveis no seguinte [Link](https://github.com/GabrielCF2/Projeto-PIBITI) - <https://github.com/GabrielCF2/Projeto-PIBITI>.

Referências

- ANDRADE, P. R. d. L.; SCARPIN, C. T.; STEINER, M. T. A. Utilização de programação linear binária para elaboração da grade horária do curso de engenharia de produção da ufpr. *Anais do XXXII Encontro Nacional de Engenharia de Produção (SBPO)*, Bento Gonçalves-RS, 2012. Citado na página 16.
- ARENALES, M. et al. *Pesquisa Operacional*. 4^a reimpressão. [S.l.]: Rio de Janeiro: Elsevier, 2007. Citado na página 20.
- BAZZAZI, M.; SAFAEI, N.; JAVADIAN, N. A genetic algorithm to solve the storage space allocation problem in a container terminal. *Computers & Industrial Engineering*, Elsevier, v. 56, n. 1, p. 44–52, 2009. Citado na página 17.
- CATTRYSSE, D. G.; WASSENHOVE, L. N. V. A survey of algorithms for the generalized assignment problem. *European journal of operational research*, Elsevier, v. 60, n. 3, p. 260–272, 1992. Citado na página 11.
- CHEN, X. et al. Design and application of an improved genetic algorithm to a class scheduling system. *International Journal of Emerging Technologies in Learning (iJET)*, International Journal of Emerging Technology in Learning, v. 16, n. 1, p. 44–59, 2021. Citado na página 18.
- DIAS, B. d. S. Modelo matemático: problema de alocação de turmas em uma instituição de ensino superior. 2018. Citado 3 vezes nas páginas 11, 16 e 20.
- DU, D. et al. *Introduction to combinatorial optimization*. [S.l.]: Springer, 2022. Citado na página 11.
- FERNANDES, K. C. C. Técnicas de otimização multiobjetivo e otimização estocástica para o roteamento de fluxos em redes. 2019. Citado 2 vezes nas páginas 11 e 13.
- FRACCAROLI, E. S. *Alocação de tarefas para a coordenação de robôs heterogêneos aplicados a agricultura de precisão*. Tese (Doutorado) — Universidade de São Paulo, 2018. Citado na página 16.
- GROTSCHER, M.; LOVÁSZ, L. Combinatorial optimization. *Handbook of combinatorics*, North-Holland Amsterdam, v. 2, n. 1541-1597, p. 4, 1995. Citado na página 11.
- HOLLAND, J. H. Genetic algorithms. *Scientific american*, JSTOR, v. 267, n. 1, p. 66–73, 1992. Citado 2 vezes nas páginas 11 e 16.
- LACHTERMACHER, G. *Pesquisa operacional na tomada de decisões*. [S.l.]: Grupo Gen-LTC, 2016. Citado 3 vezes nas páginas 13, 14 e 20.
- LAMBORA, A.; GUPTA, K.; CHOPRA, K. Genetic algorithm-a literature review. In: IEEE. *2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon)*. [S.l.], 2019. p. 380–384. Citado na página 17.

- MARINS, F. A. S. Introdução à pesquisa operacional. *São Paulo: Cultura Acadêmica: Universidade Estadual Paulista*, 2011. Citado na página 14.
- MASOUD, M. et al. Optimal assignment of e-scooter to chargers. In: IEEE. *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. [S.l.], 2019. p. 4204–4209. Citado na página 16.
- MASOUD, M. et al. A simulated annealing for optimizing assignment of e-scooters to freelance chargers. *Sustainability*, MDPI, v. 15, n. 3, p. 1869, 2023. Citado na página 16.
- MIRJALILI, S. Genetic algorithm. *Evolutionary algorithms and neural networks: theory and applications*, Springer, p. 43–55, 2019. Citado 3 vezes nas páginas 11, 16 e 17.
- MIRSADEGHI, E.; KHODAYIFAR, S. Hybridizing particle swarm optimization with simulated annealing and differential evolution. *Cluster Computing*, Springer, v. 24, p. 1135–1163, 2021. Citado na página 16.
- OLIVEIRA, E. A. M. et al. Programação linear aplicada a designação de pessoas: uma análise da literatura especializada. In: *Simpósio de Engenharia de Produção*. [S.l.: s.n.], 2019. Citado 2 vezes nas páginas 11 e 14.
- ÖNCAN, T. A survey of the generalized assignment problem and its applications. *INFOR: Information Systems and Operational Research*, Taylor & Francis, v. 45, n. 3, p. 123–141, 2007. Citado na página 14.
- SILVA, H. G. A. d. Um estudo de caso de aplicação de sistemas de recomendação baseado em filtragem colaborativa na tarefa de designação de profissionais para serviços de tic offshore. Rio das Ostras, 2020. Citado na página 16.
- SRINIVAS, M.; PATNAIK, L. M. Genetic algorithms: A survey. *computer*, IEEE, v. 27, n. 6, p. 17–26, 1994. Citado na página 17.
- TAHA, H. A. *Pesquisa Operacional. 8a Edição*. [S.l.]: São Paulo: Pearson Prentice Hall, 2008. Citado na página 14.