



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS RUSSAS
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

ANDRÉ VINICIUS SANTOS DE LIMA

**APLICAÇÃO DE TÉCNICAS DE PROGRAMAÇÃO LINEAR INTEIRA PARA
ALOCAÇÃO DE PROFESSORES E DISCIPLINAS NA UNIVERSIDADE FEDERAL
DO CEARÁ - CAMPUS RUSSAS**

RUSSAS

2022

ANDRÉ VINICIUS SANTOS DE LIMA

APLICAÇÃO DE TÉCNICAS DE PROGRAMAÇÃO LINEAR INTEIRA PARA ALOCAÇÃO
DE PROFESSORES E DISCIPLINAS NA UNIVERSIDADE FEDERAL DO CEARÁ -
CAMPUS RUSSAS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Engenharia de Software
do Campus Russas da Universidade Federal do
Ceará, como requisito parcial à obtenção do
grau de bacharel em Engenharia de Software.

Orientadora: Profa. Dra. Tatiane Fernan-
des Figueiredo

RUSSAS

2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- S233a Santos de Lima, André Vinicius.
Aplicação de técnicas de programação linear inteira para alocação de professores e disciplinas na Universidade Federal do Ceará - Campus Russas / André Vinicius Santos de Lima. – 2022.
34 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas, Curso de Engenharia de Software, Russas, 2022.
Orientação: Profa. Dra. Tatiane Fernandes Figueiredo.
1. Alocação de Aulas. 2. Programação Linear. 3. Universidade. 4. Cronograma. I. Título.
CDD 005.1
-

ANDRÉ VINICIUS SANTOS DE LIMA

APLICAÇÃO DE TÉCNICAS DE PROGRAMAÇÃO LINEAR INTEIRA PARA ALOCAÇÃO
DE PROFESSORES E DISCIPLINAS NA UNIVERSIDADE FEDERAL DO CEARÁ -
CAMPUS RUSSAS

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Engenharia de Software
do Campus Russas da Universidade Federal do
Ceará, como requisito parcial à obtenção do
grau de bacharel em Engenharia de Software.

Aprovada em: 13/12/2022

BANCA EXAMINADORA

Profa. Dra. Tatiane Fernandes Figueiredo
(Orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Marcio Costa Santos
Universidade Federal do Ceará (UFC)

Prof. Dr. Pablo Luiz Braga Soares
Universidade Federal do Ceará (UFC)

À minha família, por sua capacidade de acreditar em mim e investir em mim. Mãe, seu cuidado e dedicação foi que deram, em alguns momentos, a esperança para seguir. Pai, sua presença significou segurança e certeza de que não estou sozinho nessa caminhada.

AGRADECIMENTOS

Aos meus pais, meu irmão e a Joyce Sousa por sempre terem me apoiado na minha jornada na faculdade, tanto de maneira emocional quanto de maneira financeira.

Agradeço a todos os professores por me fornecer conhecimentos durante todas as cadeiras, e ainda proporcionarem aprendizados sobre o mercado de trabalho, e guias de pontos de partida em uma carreira desse ramo.

A Profa. Dra. Tatiane Fernandes Figueiredo por me orientar na produção deste trabalho, e me apresentar a área de Otimização.

Aos professores Dr. Bonfim Amaro Júnior e Dr. Marcio Costa Santos, os quais me deram orientações sobre a confecção deste documento, e me apresentaram a ferramenta \LaTeX .

Ao coordenador do curso Dr. Anderson Feitoza, por me proporcionar informações essenciais para a construção dos modelos do problema em estudo.

Aos meus amigos Francisco Eudes, Isaac Freitas, José Ulisses, Lucas Emanuel, Luís Gustavo Girão, Jeferson Ribeiro, Vicente Augusto e Vitória Helen, os quais estiveram comigo e me acompanharam nos estudos desde o meu primeiro semestre na faculdade.

Ao Lucas Chaves, Lucas Santiago e Hugo Venâncio, meus grande companheiros de apartamento que estiveram na minha companhia durante toda essa jornada.

Aos meus colegas de trabalho Josué Lamec, Matheus Santos, Pedro Antônio Ortega e Vinicius Strumillo, que sempre me apoiaram tanto no trabalho quando na faculdade.

“Sonhos determinam o que você quer. Ação determina o que você conquista.”

(Aldo Novak)

RESUMO

A cada semestre, os coordenadores da Universidade Federal do Ceará - Campus Russas precisam alocar seus docentes para as disciplinas de cada curso de graduação. Para isso, os coordenadores devem levar em consideração os dias e horários das disciplinas e algumas outras restrições.

A cada semestre, os coordenadores da Universidade Federal do Ceará - Campus Russas enfrentam o desafio de alocar seus docentes para as disciplinas de cada curso de graduação. Esse processo de alocação exige uma análise cuidadosa dos dias e horários das disciplinas, além de outras restrições importantes, como a carga horária dos docentes, suas áreas de especialização e as preferências individuais. Considerando a complexidade dessa tarefa, este trabalho propõe um modelo de Programação Linear Inteira (PLI) como solução para otimizar o processo de alocação. A utilização desse modelo permite lidar de forma eficiente com as diversas variáveis envolvidas, garantindo que as disciplinas sejam distribuídas de maneira equilibrada e atendendo às necessidades institucionais. Os resultados obtidos por meio de testes computacionais indicam que é possível resolver a alocação de forma ótima em um curto espaço de tempo, o facilitando a tarefa de alocação e mantendo a satisfação dos docentes envolvidos no processo.

Palavras-chave: alocação de aulas; programação linear; universidade; cronograma.

ABSTRACT

Each semester, the coordinators at the Federal University of Ceará - Russas Campus face the challenge of assigning their faculty members to the courses offered in each undergraduate program. This allocation process requires careful analysis of the days and times of the courses, as well as other important constraints, such as the faculty members' workloads, areas of expertise, and individual preferences. Given the complexity of this task, this work proposes an Integer Linear Programming (ILP) model as a solution to optimize the allocation process. The use of this model allows for efficient management of the various variables involved, ensuring that courses are distributed in a balanced manner and that institutional needs are met. The results obtained through computational tests indicate that it is possible to solve the allocation optimally in a short period of time, facilitating the allocation task and maintaining the satisfaction of the faculty members involved in the process.

Keywords: classes allocation; linear programming; university; schedule.

LISTA DE FIGURAS

Figura 1 – Subdivisões do Timetabling	16
Figura 2 – Exemplo de Branch and Bound	20
Figura 3 – Ciclo da Metodologia	24
Figura 4 – Amostra da primeira instância considerando disciplinas da Ciência da Computação (CC) e Engenharia de Software (ES)	27
Figura 5 – Informações sobre as soluções obtidas.	30
Figura 6 – Solução obtida com 16 professores, onde estes são representados por letras maiúsculas na primeira coluna.	30

LISTA DE ABREVIATURAS E SIGLAS

B&B	<i>Branch and Bound</i>
IES	Instituição de Ensino Superior
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
MEC	Ministério da Educação
NDE	Núcleo Docente Estruturante
PAAS	Problema de Alocação de Aulas a Salas
PAS	Problema de Alocação de Aulas
PL	Programação Linear
PLB	Programação Linear Binária
PLI	Programação Linear Inteira
PLIM	Programação Linear Inteira Mista
UFC	Universidade Federal do Ceará

SUMÁRIO

1	INTRODUÇÃO	12
2	OBJETIVOS	14
2.1	Objetivo geral	14
2.2	Objetivo específicos	14
3	FUNDAMENTAÇÃO TEÓRICA	15
3.1	Problemas Combinatórios	15
3.1.1	<i>Timetabling</i>	15
3.2	Programação Linear	16
3.2.1	<i>Programação Linear Inteira</i>	18
3.2.2	<i>Algoritmos para resolução de PL, PLI e PLB</i>	18
4	TRABALHOS RELACIONADOS	21
4.1	Uma abordagem de problema de alocação de professores em disciplinas utilizando programação linear	21
4.2	Um problema de programação linear para solução do problema da aplicação de recursos em atividades complementares em instituições de ensino superior	22
4.3	Modelo de programação linear inteira para o problema de alocação de salas: estudo de caso em uma Instituição de Ensino Superior	22
5	METODOLOGIA	24
5.1	Coleta de informações do problema	24
5.2	Formação de uma base de dados para o problema	26
5.3	Criação de um modelo PLI para o problema	27
5.4	Geração de soluções	29
6	CONCLUSÕES E TRABALHOS FUTUROS	32
6.1	Considerações gerais	32
6.2	Trabalhos Futuros	33
	REFERÊNCIAS	34

1 INTRODUÇÃO

Todos os semestres, nas instituições de ensino superior, surge a necessidade da criação de uma alocação de disciplinas ofertadas para um conjunto de professores. Porém, a tarefa de alocar professores às disciplinas e distribuí-las em dias e horários, seguindo um conjunto de restrições definidos por cada instituição, é uma operação, na grande maioria das vezes, feita manualmente. Este tipo de tarefa pode ser muito complexa, uma vez que o número de possibilidades de combinações, dependendo da quantidade de professores e disciplina, pode ser muito grande, impossibilitando a análise completa de todos os casos. Problemas semelhantes a este, são conhecidos na literatura como Problema de Alocação de Aulas (PAS) e que por ter um padrão combinatório, são classificados na literatura de problemas NP-Difíceis.

Embora os modelos matemáticos existentes para resolução de problemas da classe NP-Difícil tendam a não solucionar instâncias suficientemente grandes em um tempo considerado viável, alguns modelos para problemas de alocação podem obter soluções ótimas ou boas soluções, com um gasto de tempo aceitável, dependendo da entrada. Uma possível forma é a criação de modelos de PLI, uma técnica de otimização que visa criar um modelo matemático representativo de um problema utilizando apenas equações e inequações lineares, restringindo o domínio das variáveis apenas em valores inteiros.

É possível encontrar trabalhos na literatura que fazem uso de modelos de PLI para gerar soluções para problemas de alocação. Filho (2016) apresenta um modelo de PLI com foco em solucionar o problema de construção de cronograma buscando maximizar a relação dos professores com suas disciplinas. Da mesma forma, Lara (2004) busca solucionar o mesmo impasse, dando mais atenção para a questão econômica do problema. Já no trabalho do Silva *et al.* (2019), o problema é solucionado com um modelo de PLI dedicado a alocar disciplinas às salas.

Considerando a problemática da Universidade Federal do Ceará (UFC) - Campus Russas, este trabalho apresenta um modelo de PLI que soluciona o problema de alocação de professores às disciplinas, dias e horários conforme requisitado pelas coordenações dos cursos de Ciência da Computação e Engenharia de Software. O modelo apresentado nesta monografia considera algumas restrições específicas da instituição, assim como a satisfação dos professores envolvidos. Os resultados obtidos através de testes computacionais utilizando uma instância real comprovam a eficiência do modelo em gerar uma solução para o problema proposto.

Este trabalho está constituído da seguinte forma: no Capítulo 2 é apresentado o

objetivo geral e os objetivos específicos, almejados neste trabalho; no Capítulo 3 são mostrados os conhecimentos utilizados para a realização do estudo do problema proposto; no Capítulo 4 é exposto os trabalhos semelhantes, os quais também visam solucionar o problema de alocação de disciplinas em universidades. No Capítulo 5 é apresentado a metodologia dos processos executados para o desenvolvimento desta pesquisa. Por fim, o Capítulo 6 apresenta as conclusões finais e possíveis trabalhos futuros.

2 OBJETIVOS

Neste capítulo será abordado o objetivo principal a ser atingido com o desenvolvimento deste trabalho, além de metas necessárias para a obtenção dos resultados.

2.1 Objetivo geral

Criação de um modelo de Programação Linear Inteira para resolução do problema de alocação de professores à disciplinas na UFC - Campus Russas.

2.2 Objetivo específicos

- Realizar entrevistas com *stakeholders* da UFC - Campus Russas quem tenham relação direta ou indireta com o problema;
- Documentar as informações importantes retiradas de reuniões com *stakeholders* da UFC - Campus Russas;
- Gerar um base de dados relacionada a temática;
- Estudar problemas semelhantes existentes na literatura;
- Criar um modelo de PLI que possam solucionar o problema;
- Definir qual linguagem e *framework* será utilizado na etapa de implementação;
- Implementar o de modelo de PLI criado e realizar testes;
- Analisar os resultados obtidos.

3 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos da literatura usados para a construção de modelos de PLI para resolução de problemas combinatórios. A subseção 3.1 apresenta uma pequena introdução sobre problemas combinatórios, além de expor o que é o impasse do *Timetabling* (3.1.1), junto de suas possíveis formas de interpretá-lo, e obter soluções viáveis para o mesmo, enquanto a subseção 3.2 descreve sobre como solucionar problemas combinatórios utilizando PLI e suas subdivisões.

3.1 Problemas Combinatórios

Problemas combinatórios, englobam questões onde se tem um conjunto de elementos, os quais são relacionados a um grupo de regras. Em geral, neste tipo de problema é permitido a criação de novos subconjuntos, a partir da escolha de determinados elementos respeitando uma determinada regra. Note que o espaço de solução, dependendo do tamanho do conjunto de elementos, pode ser muito grande, ou seja, um problema deste tipo pode conter uma explosão combinatória de soluções. Portanto, sendo inviável o uso de algoritmos simples para sua resolução, como enumerar todos as possíveis soluções do espaço, segundo Aparecido (2022).

Em geral, este tipo de problema é classificados como NP-Difícil, estando presentes em impasses bastante comuns no cotidiano, como casos de problemas de empacotamento de produtos, roteirizar a entrega de cargas de caminhões de uma fábrica, escalonar tarefas a máquinas de uma indústria, ou até casos mais complexos como o Projeto GENOMA, o qual visa a busca de soluções para problemas de Biologia Molecular.

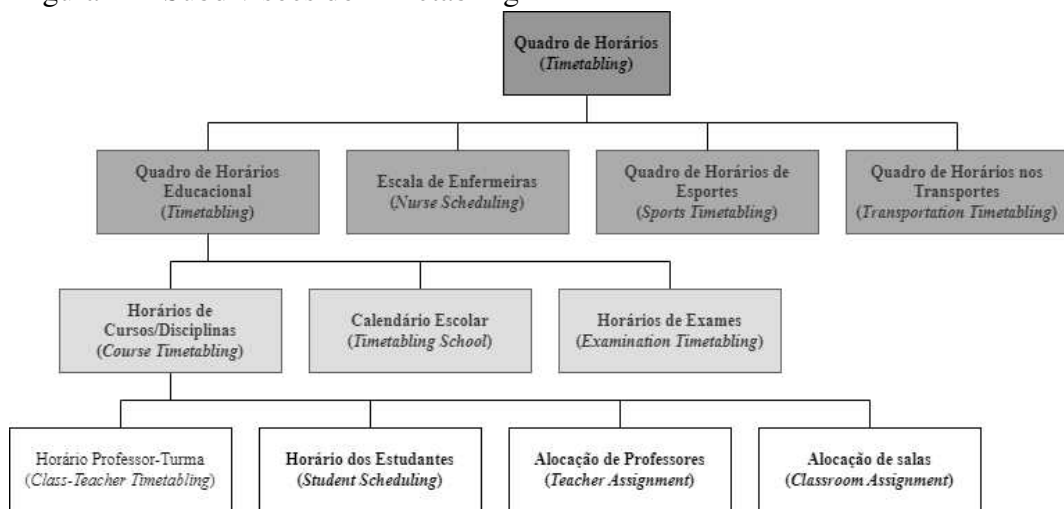
Neste trabalho, será abordado o problema de alocação de horários no contexto da Universidade Federal do Ceará - Campus Russas. Este tipo de problema pode ser classificado como *Timetabling*, um problema de natureza combinatória.

3.1.1 *Timetabling*

O problema de alocação de horários e aulas e a criação de um cronograma acadêmico, é uma questão combinatória da literatura conhecido por *Timetabling*, possuindo várias ramificações fundamentadas nas diversas formas de interpretar este problema e nas divergências entre os modelos originados por conta das diversas variedades de restrições no sistema educacional das instituições.

Assim, o problema de *Timetabling* não possui um modelo universal para sua resolução e pode ser tipificado de acordo com as restrições do problema em estudo. Um exemplo é o *Teacher Assignment*, o qual é mais focado em solucionar um sistema mais similar ao que é adotado nas instituições de ensino superior, visto que cada aluno pode escolher quais disciplinas irá cursar durante um semestre. Já o *Class-Teacher*, aborda o problema em que uma turma terá disciplinas já pré-designadas. Neste caso os modelos apresentados na literatura tratam de buscar uma solução ótima para alocar seu corpo docente. Outra forma de se observar o problema, é analisar a alocação de salas para cada disciplina da instituição, buscando impedir o choque de horários entre as disciplinas dadas em mesma sala, como foi citado por Lara (2004).

Figura 1 – Subdivisões do Timetabling



Fonte: Adaptado de Sales *et al.* (2015).

Todas as versões apresentadas acima do problema de *Timetabling* acima são NP-Difícil, havendo na literatura diversos artigos que apresentam técnicas como PLI, aplicação de algoritmos genéticos e do uso de heurísticas para sua resolução, segundo Silva *et al.* (2019).

3.2 Programação Linear

A Programação Linear (PL) se enquadra em uma das áreas da Pesquisa Operacional e tem sido amplamente utilizada para resolução de problemas combinatórios. Modelos de PL são utilizados para resolução de diversos tipos de problemas científicos e aplicações do mundo real. Como exemplo, pode-se citar problemas na agricultura, mineração, siderúrgicas, criação de rotas, dentre outros. Em suma, é possível afirmar que esta técnica fornece a maximização, ou minimização de uma função objetivo expressa de forma linear, garantindo que sejam satisfeitas

um conjunto de restrições também expressas por equações ou inequações lineares (PRADO, 2016).

Mais precisamente, modelos de PL são representações abstratas de um problema, o qual suas restrições são definidas por conjunto de equações e/ou inequações formadas por variáveis de decisão multiplicadas por valores constantes (designados por parâmetros do modelo), como é mostrado na representação a seguir:

$$\begin{aligned} a_{1,1} * x_1 + a_{2,1} * x_2 + \dots + a_{n,1} * x_n &< b_1 \\ a_{1,2} * x_1 + a_{2,2} * x_2 + \dots + a_{n,2} * x_n &< b_2 \\ &\dots \\ a_{1,m} * x_1 + a_{2,m} * x_2 + \dots + a_{n,m} * x_n &< b_m \end{aligned}$$

Para o conjunto de restrições lineares acima, as variáveis de decisão $\{x_1, x_2, \dots, x_n\}$ assumem valores reais, garantindo que sejam satisfeitos todos os limites definidos pelas constantes $\{b_1, b_2, \dots, b_m\}$, considerando cada constante $\{a_{1,1}, a_{1,2}, \dots, a_{1,n}, \dots, a_{n,m}\}$.

Outro ponto a ser destacado sobre as restrições do modelo, é a possibilidade da existência de restrições de não-negatividade. Estas são responsáveis por limitar que os insumos não assumam quantidades negativas. Muitos problemas do mundo real necessitam deste tipo de restrição, como por exemplo, problemas de produção cujo não existe a possibilidade dos insumos possuir uma quantidade negativa. A seguir é apresentado, o conjunto de restrição de não-negatividade para suas variáveis descritas acima:

$$x_1, x_2, \dots, x_n \geq 0$$

Por fim, o valor dado pela função objetivo de um problema de PL é usualmente representado pela letra Z . A função objetivo tem o propósito de buscar a melhor solução para o problema, também denominada como valor ótimo. Como exemplo de aplicação, uma função objetivo pode visar a maximização de um lucro, diminuir os custos, evitar desperdícios, ou melhorar a eficiência do uso dos alguns recursos, segundo Carvalho (2014). A função objetivo, considerando as variáveis descritas na restrição anterior, é apresentada da seguinte forma:

$$\text{MAX ou MIN } Z = c_1 * x_1 + c_2 * x_2 + \dots + c_n * x_n$$

Neste exemplo, o MAX e MIN definem se a função buscará o maior, ou o menor valor para Z . As constantes $\{c_1, c_2, \dots, c_n\}$ representam os valores dos parâmetros do modelo de PL, os quais serão multiplicados pela variáveis de decisão do modelo.

3.2.1 Programação Linear Inteira

Os problemas de PLI são vistos como uma ramificação da PL, que abordam problemas de otimização onde as variáveis de decisão não podem assumir valores contínuos, sendo restringidas apenas a valores inteiros. A seguir é apresentada o conjunto de restrições que devem ser adicionados as variáveis do exemplo da Seção anterior para que um modelo de PL se torne um modelo de PLI.

$$x_1, x_2, \dots, x_n \in \mathbb{Z}$$

Caso essas restrições não se apliquem a todas as variáveis, o problema será designado por problema de Programação Linear Inteira Mista (PLIM). Por fim, os problemas também podem ser avaliados como Programação Linear Binária (PLB). Como exemplo de aplicação, podem ser citadas questões onde as variáveis de decisão são responsáveis por fazer a escolha de usar, ou não um devido recurso. Portanto, para representar este requisito em um modelo de PL se faz necessário a adição de uma restrição que obriga as variáveis de decisão assumirem apenas os valores de 0 ou 1, como citado por Wolsey (2021). A seguir é apresentada o conjunto de restrições que devem ser adicionados ao exemplo da Seção anterior para que o modelo de PL seja classificado como PLB:

$$x_1, x_2, \dots, x_n \in \{0, 1\}$$

3.2.2 Algoritmos para resolução de PL, PLI e PLB

Os Problemas de Programação Linear podem ser facilmente solucionados com a execução do algoritmo *Simplex*, de maneira pseudo-polinomial. No entanto, ao tratar de problemas PLI e PLB, apenas a aplicação do algoritmo *Simplex* nem sempre é o bastante. Tendo em vista a robustez destes problemas, pode se fazer necessário o uso de técnicas mais capacitadas para solucionar estes impasses, como exemplo pode-se citar a técnica de *Branch and Bound*, por ser uma das mais comumente utilizadas.

O algoritmo de *Branch and Bound* (B&B) visa solucionar problemas de PLI e PLB com base na estratégia de dividir e conquistar, assim dividindo o problema geral em partes menores com intuito de encontrar apenas soluções viáveis, incluindo a solução ótima. As divisões do problema podem ser representadas por uma árvore, a qual teria como raiz o problema geral a ser solucionado, e as ramificações seriam seus subproblemas.

Cada subproblemas é solucionado por meio do relaxamento das variáveis, o qual é retira as restrições de integralidade, e em seguida é executado o algoritmo *Simplex* para obtenção da solução do subproblema relaxado, ou a verificação que não existe solução para o mesmo. No caso da solução ótima conter valores contínuos, é realizada a escolha de uma delas para ser ramificada, gerando dois novos subproblemas com restrições de limites inferior, ou superior para a variável escolhida.

Come é citado por Brusco e Stahl (2006), o B&B utiliza vários componentes na sua busca pela solução, porém destes temos alguns com maior relevância, os quais podemos citar:

- *branching*: este passo pode ser considerado como o ato de ramificar a árvore representante do problema, onde ocorrerá o relaxamento do problema, e uma das variáveis que obter um valor contínuo na solução, será escolhida para ser expandida.
- *bouding*: esta é a etapa de designar os limites inferiores e superiores da variável a ser expandida em um subproblema, onde esta limitação é designada de acordo com o valor inteiro mais próximo, acima e a baixo da variável continua em expansão.
- *pruning*: também podendo ser entendido como o podamento da árvore, esta etapa visa parar a expansão de um dos ramos quando um problema é inviável, ou é encontrado um solução viável para o problema geral, e também há o caso de ser obtido uma solução ótima para um subproblema, a qual seja igual ou pior que o valor resultado em outro ramo.
- *retracting*: esta etapa ocorrerá logo após o *pruning*, assim ao ocorrer o podamento da árvore será feito um retorno na profundidade da mesma, para explorar as soluções de subproblemas ainda não analisados.

Por exemplo, na figura 2 é possível ver a árvore gerada por um B&B ao tentar solucionar o seguinte modelo:

$$\text{Função Objetivo: } \text{MAX } Z = x_1 + 4x_2 \tag{3.1}$$

$$\text{Restrições: } S.A - 2x_1 + 4x_2 < 8 \tag{3.2}$$

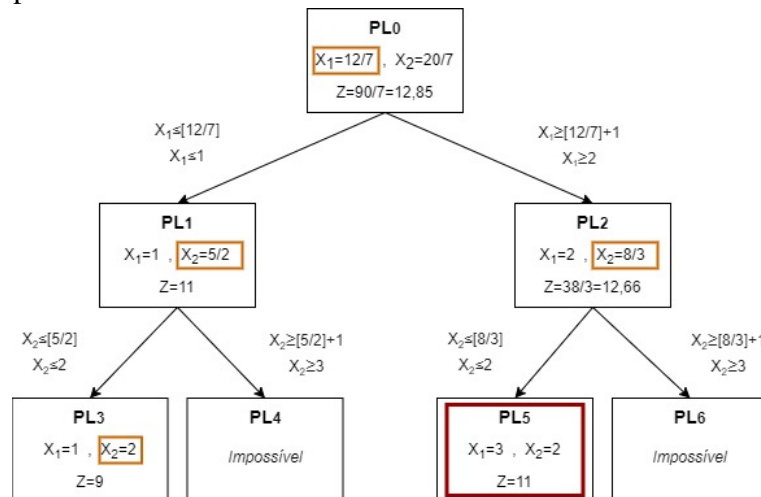
$$2x_1 + 3x_2 < 12 \tag{3.3}$$

$$x_1, x_2 \leq 0 \tag{3.4}$$

$$x_1, x_2 \in \mathbb{Z} \tag{3.5}$$

Na árvore da solução, o PL_0 representa a raiz do problema em análise, o qual será ramificado sempre que X_1 , ou X_2 assumirem valores não inteiros, devido o relaxamento das variáveis, como ocorre em PL_1 e PL_2 . Outro ponto notável são as folhas PL_3 e PL_5 , as quais são soluções plausíveis para o problema, uma vez que suas variáveis assumiram valores inteiros, mesmo com o relaxamento das restrições. Nesse sentido, o PL_5 é responsável por apresentar a solução ótima do problema, devido Z ter obtido o maior resultado para o problema de maximização em análise.

Figura 2 – Exemplo de Branch and Bound



Fonte: Criado pelo autor.

Na árvore anterior, a cada problema em análise é feito o relaxamento das variáveis, e executado um *Simplex*, e escolhido uma das variáveis as quais resultaram em valores não inteiros

4 TRABALHOS RELACIONADOS

Este capítulo apresenta trabalhos semelhantes ao atual, onde todos também abordam problemas de alocação de atividades em Instituições de Ensino Superior, contudo havendo diferenças na forma de tratar o problema. Na seção 4.1 é mostrado uma forma de construir um modelo visando maximizar a afinidade dos professores com as disciplinas designadas aos mesmos. Na seção 4.2 é apresentado um ponto de vista mais econômico acerca da alocação de atividades complementares em Instituições de Ensino Superior. Na seção 4.3 é exposto uma solução para o problema de *timetabling* focado na parte de alocação de salas.

4.1 Uma abordagem de problema de alocação de professores em disciplinas utilizando programação linear

O autor Filho (2016) apresenta a formulação de um modelo fundamentado na programação linear inteira, o qual visa resolver o Problema de Alocação de Professores em Disciplinas para a Universidade Federal do Ceará - Campus Quixadá. Assim, o autor tem por objetivo fornecer uma ferramenta que facilite a tarefa de alocação ainda almejando o melhor resultado possível, dentro das restrições da instituição.

Para a implementação deste modelo, o autor do trabalho considera as restrições informadas pelo o Campus de Quixadá, e também o grau de afinidade dos professores com a disciplina a ser atribuída a ele. A função do modelo teve por objetivo, maximizar a vinculação dos professores com as disciplinas, se mantendo dentro das limitações informadas pela Universidade.

Para analisar as soluções obtidas pelo modelo, Filho (2016) utilizou instâncias com dados aleatórios e instancias com dados reais. No experimento com dados reais, a afinidade dos professores com a disciplina, foram fundamentadas no número de vezes que o professor ensinou a disciplina, e ao realizar uma comparação entre a alocação do modelo e a alocação empregada na UFC-Quixadá em 2016.2, o valor de função do modelo se mostrou 20% mais eficiente. A alocação empregada pela própria Universidade no semestre estudado obteve um total de 394 pontos, enquanto o modelo gerou uma solução que resultou em 475 pontos. No teste com dados aleatórios, as conclusões atingidas pelo autor, foram que a quantidade de preferências atribuídas exerce um impacto tanto no tempo de resposta, quanto na possibilidade de existência de uma solução para o problema. Também foi notório que o modelo obteve bons resultados até para instâncias com números de professores e disciplinas, maiores que a da instituição.

4.2 Um problema de programação linear para solução do problema da aplicação de recursos em atividades complementares em instituições de ensino superior

Lara (2004) aborda o problema de *timetabling*, mostrando suas variações, e busca realizar a criação e aplicação de um modelo linear para a resolução do problema de alocação de atividades complementares ao corpo docente das Instituições de Ensino Superior, visando alcançar o custo mínimo, mantendo a eficiência dos recursos e investimentos.

Os primeiros modelos gerados pelo autor foram fundamentado nos critérios de avaliação do Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), o qual é aplicado pelo Ministério da Educação (MEC). Portanto a função objetivo busca o menor custeio para as atividades complementares, possibilitando um bom uso do orçamento da instituição, garantindo que a mesma seja bem avaliada pelo INEP. O autor constatou em seu trabalho que o primeiro modelo proposto, embora tenha retornado soluções ótimas, se tornou inviável na prática para a Instituição de Ensino Superior (IES). Em vista disso o autor decidiu ampliar o modelo considerando o conjunto de qualificações de cada professor membro da instituição, considerando a diferença salarial de acordo com o grau de titulações.

Já na análise do modelo final proposto, Lara (2004) chegou a conclusão que o mesmo ignora o diagnóstico de alguns fatores, como a CLT e a escassez de mão de obra, além de não operar na camada operacional, onde é tratado questões de alocação de professores em horários de maneira efetiva, evitando conflitos. Contudo o modelo opera no nível estratégico, fornecendo informações aos gestores acerca dos custos de alocação de recursos e a carga horaria de atividades complementares. Desta forma, foi concluído que o modelo pode auxiliar a instituição oferecendo orientações na construção de uma política de alocação de carga horária complementar.

4.3 Modelo de programação linear inteira para o problema de alocação de salas: estudo de caso em uma Instituição de Ensino Superior

Silva *et al.* (2019) realizou um estudo acerca do PAS, ou Problema de Alocação de Aulas a Salas (PAAS), uma das ramificações do problema de *educational timetabling*, que visa a alocação eficiente de disciplinas a salas, considerando as capacidades de cada sala e não permitindo o uso para duas disciplinas no mesmo horário. O objetivo seria construir um modelo que otimizasse a alocação de salas para o Centro de Informática da Universidade Federal da Paraíba, buscando ainda diminuir o tempo usualmente gasto para a resolução deste problema.

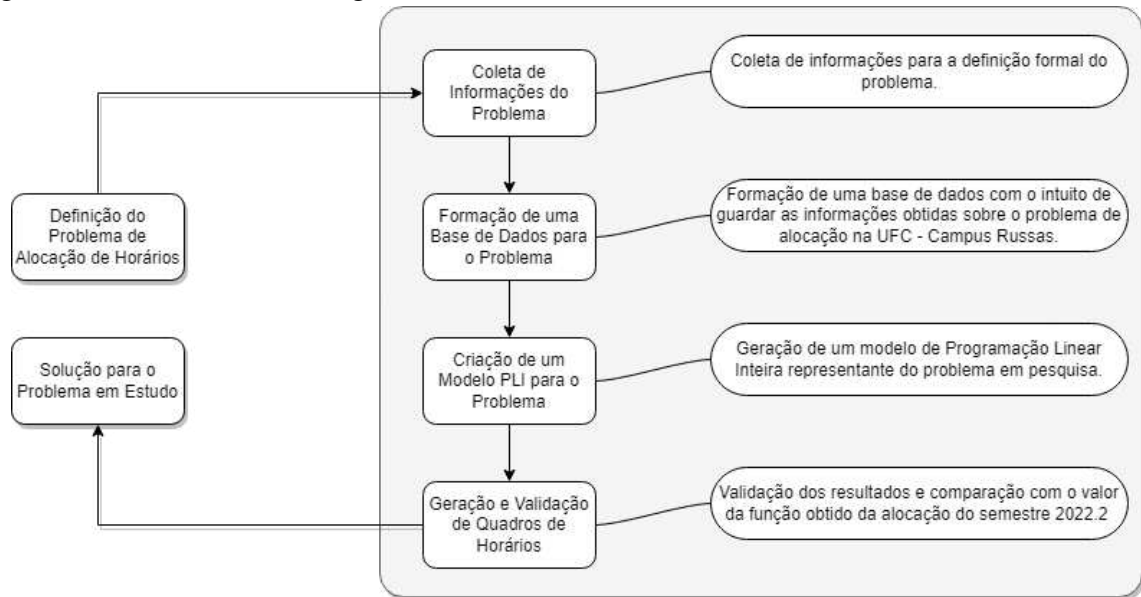
O estudo feito foi fundamentado nas restrições do Centro de Informática da UFPB. A função objetiva do problema, buscava minimizar o número de aulas de uma mesma turma alocada em salas diferentes, também considerando um custo fundamentado no fator de preferência ao designar uma determinada sala a uma disciplina, e realizando uma punição caso haja excesso de demanda. tendo por limitações o impedimento de duas disciplinas serem lecionadas na mesma sala e horário, a restrição de uma disciplina não poder ser dada em salas diferentes no mesmo horário, além do controle de uma aula não poder ser alocada em uma sala a qual não suportará a carga de alunos matriculados.

Nos resultados o autor Silva *et al.* (2019) realizou comparações com as alocações feitas no semestre de 2019.1, o qual ofertou 103 disciplinas a serem alocadas em 17 salas, com um horário semanal das 08:00 as 17:30 de segunda a sexta. Foi notado que a alocação real continha vários casos de falta de homogeneidade, as quais apontavam uma má disponibilização dos recursos. Com estas instâncias obtidas, foram feitas duas execuções no modelo, onde no primeiro resultado ocorreu a obtenção de uma organização em que apenas duas disciplinas tiveram suas aulas alocadas em duas salas diferentes, e o uso do auditório passou de 5 turmas para apenas 2. No segundo resultado, foi solicitado pela gestão que não houvesse uso do auditório, porém ocorresse mais alocações nas salas de um dos prédios da instituição, e por consequência foi atingido um resultado que houve uma melhor distribuição das disciplinas.

5 METODOLOGIA

Nesta capítulo é definida a metodologia utilizada com a finalidade de alcançar o objetivo central desta pesquisa. A Figura 3 apresenta o fluxo dos processos realizados, onde cada uma das etapas definidas estão detalhadas em seções específicas a seguir.

Figura 3 – Ciclo da Metodologia



Fonte: Criado pelo autor.

5.1 Coleta de informações do problema

As informações necessárias para entender as necessidades do problema e suas restrições, foram obtidas por meio de um entrevista com o coordenador do curso de Engenharia de Software da UFC - Campus Russas. A entrevista foi fundamentada em torno de onze perguntas realizadas ao coordenador do curso, as quais abordaram questões acerca da alocação de horários por meio perguntas sobre a carga horária dos professores, fatores influentes na alocação de professores a disciplinas, e dinâmicas na distribuição das disciplinas aos horários. Abaixo é apresentado as onze perguntas utilizadas na entrevista:

1. Um professor tem uma quantia máxima de aulas que ele pode dar em um dia?
A
2. Quantas horas de aula um professor deve dar por semana?
3. Fora de aula, um professor deve permanecer no campus, por pelo menos quantas

horas?

4. Ainda no mesmo contexto, existe algum limite mínimo de horas de permanência por dia?
5. Em quais casos é necessário que sejam alocados dois professores para uma mesma cadeira?
6. Como é definida a escolha das disciplinas de cada professor? Existem critérios, quais? Exemplos: utilizar dados históricos, dados sobre a especialidade de cada professor, taxa de aprovação (onde eu posso obter esses dados) e grau de satisfação do professor em lecionar cada disciplina (perguntar para o professor). Existe mais alguma consideração por parte da coordenação.
7. Existe regra para alocar disciplinas iguais para um mesmo professor? É melhor para o curso que um mesmo professor dê disciplinas idênticas para cursos distintos ou não? Exemplo: Projeto e Análise de Algoritmos é dada para Ciência da Computação e Engenharia de Software, seria melhor o mesmo professor ou dois professores distintos?
8. Quais são os critérios para definir os horários de cada turma em cada semestre? Exemplo: Uma turma deve ter seus horários espalhados ou condensados? Se for horário condensado seria melhor manhã ou tarde em quais circunstâncias?
9. Seria interessante considerar o grau de complexidade da disciplina para alocar seu horário. Exemplo: Disciplinas mais complexas no início do turno e disciplinas mais leves no final do turno.
10. Seria interessante não alocar o mesmo professor para duas disciplinas de uma mesma turma. Exemplo: Laboratório de Programação e Estrutura de Dados são disciplinas da turma de Ciência da Computação e Engenharia de Software do segundo semestre. Seria algo negativo ser dado pelo mesmo professor para a coordenação. Na visão do aluno, aparentemente não é interessante.
11. Seria interessante tentar espalhar os horários de aula em um mesmo dia do professor ou condensá-las (seria interessante perguntar para os professores)?

Ao fim da entrevista, foram obtidas respostas para todas as onze perguntas realizadas, onde foi possível captar informações sobre o padrões na determinação dos horários por semestre (um exemplo é a prática de em um semestre alocar matérias prioritariamente pela manhã, e no

próximo focar em destinar as cadeiras para o período da tarde), e entender que algumas questões estão mais relacionadas a preferência dos professores.

5.2 Formação de uma base de dados para o problema

A base de dados utilizada para testar o modelo proposto e gerar a alocação requerida pelas coordenações dos cursos foi criada considerando os graus de afinidade de cada professor com cada uma das disciplinas dos cursos e dados fornecidos pelas secretarias e Núcleo Docente Estruturante (NDE) dos cursos de Ciência da Computação e Engenharia de Software. Desta forma, a base de dados é composta por:

- Número de salas;
- Disciplinas que devem ser obrigatoriamente ofertadas em cada semestre e disciplinas optativas;
- Grau de afinidade dos professores para lecionar cada uma das disciplinas ofertadas em cada semestre.

De posse da base de dados, foi possível a formação de uma instância para o problema de alocação de professores na UFC - Campus Russas, considerando o semestre 2022.2. A instância criada possui um total de 15 salas destinadas a aulas dos cursos de Ciência da Computação e Engenharia de Software, 16 professores e 44 disciplinas. O grau de afinidade do professor com cada disciplina foi representado através de uma ponderação inteira entre 0 e 3, onde quanto maior o valor, maior a afinidade do professor com a disciplina. A Figura 3 apresenta um exemplo da ponderação criada.

Figura 4 – Amostra da primeira instância considerando disciplinas da Ciência da Computação (CC) e Engenharia de Software (ES)

Disciplinas	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Ética Profissional ES	0	0	0	1	0	0	0	0	2	1	0	1	0	1	1	1
Ética Profissional CC	0	0	0	1	0	0	0	0	2	1	0	1	0	1	1	1
Introdução à Engenharia de Software ES	0	3	0	1	1	0	3	0	3	3	0	1	1	2	1	3
Fundamentos de Programação ES	2	0	0	3	2	2	0	2	3	1	2	1	3	2	3	0
Introdução à Ciência da Computação CC	2	0	1	2	1	2	0	3	1	2	1	1	3	2	1	2
Arquitetura de Computadores ES	0	0	0	1	0	2	0	0	0	0	1	1	1	1	0	0
Arquitetura de Computadores CC	0	0	0	1	0	2	0	0	0	0	1	1	1	1	0	0
Introdução a Processos e Requisitos de Software ES	0	3	0	0	0	0	3	0	3	3	0	3	0	0	0	3
Laboratório de Programação ES	0	0	0	3	2	2	0	2	3	0	1	0	2	0	0	0
Laboratório de Programação CC	0	0	0	3	2	2	0	2	3	0	1	0	2	0	0	0
Estruturas de Dados ES	2	0	0	2	3	2	0	3	0	0	1	0	2	1	3	0
Estruturas de Dados CC	2	0	0	2	3	2	0	3	0	0	1	0	2	1	3	0
Sistemas Operacionais CC/ES	2	0	0	2	0	2	0	0	0	0	1	0	1	1	0	0
Programação Orientada a Objetos CC/ES	2	0	1	3	0	2	0	2	3	0	2	1	0	2	3	0
Estrutura de Dados Avançada CC	2	0	0	1	3	0	0	3	0	0	1	0	1	0	3	0
Linguagens Formais e Autômatos	3	0	3	1	2	0	0	0	0	0	1	0	2	0	0	0
Linguagens de Programação CC/ES	0	0	0	2	0	0	0	3	1	0	1	0	0	2	0	0
Requisitos de Software ES	0	3	0	0	0	0	3	0	3	2	0	2	0	0	0	3
Algoritmos em Grafos ES	3	0	0	0	0	0	0	2	0	0	0	0	3	0	0	0
Algoritmos em Grafos CC	3	0	0	0	0	0	0	2	0	0	0	0	3	0	0	0

Fonte: Criado pelo autor.

5.3 Criação de um modelo PLI para o problema

Seja a entrada do problema de alocação de professores e disciplinas da UFC-Campus Russas uma 6-tupla definida como $\beta = (A, H, P, S, D, f)$, onde A é conjunto de pares de dias da semana que uma disciplina pode ser alocada, H o conjunto de horários possíveis para uma disciplina ser ofertada, P e S o conjunto de professores e salas disponíveis respectivamente, D o conjunto de disciplinas que devem ser ofertadas no semestre vigente e $f : P \times D \rightarrow \{0, 1, 2, 3\}$ uma função que retorna o grau de afinidade de um professor com uma disciplina, o seguinte modelo de PLI é apresentado para resolução da problemática proposta.

Variáveis definidas:

$$W_{i,j,k,l} = \begin{cases} 1, & \text{se a disciplina } j \text{ foi alocada no horário } k \text{ e nos dias } l \text{ para o professor } i \\ 0, & \text{caso contrário.} \end{cases} \quad (5.1)$$

Modelo proposto:

$$\max Z = \sum_{i \in P} \sum_{j \in D} \sum_{k \in H} \sum_{l \in A} f(i, j) W_{i,j,k,l} + p_1 + p_2 \quad (5.2)$$

$$p_1 = \sum_{i \in P} (3 - \sum_{j \in D} \sum_{k \in H} \sum_{l \in A} W_{i,j,k,l}) \quad (5.3)$$

$$p_2 = \sum_{k \in H} \sum_{l \in A} \left(\frac{S}{2} - \sum_{i \in P} \sum_{j \in D} W_{i,j,k,l} \right) \quad (5.4)$$

$$\text{s.a: } \sum_{j \in D} \sum_{k \in H} \sum_{l \in A} W_{i,j,k,l} \leq 4 \quad \forall i \in P \quad (5.5)$$

$$\sum_{j \in D} \sum_{k \in H} \sum_{l \in A} W_{i,j,k,l} \geq 2 \quad \forall i \in P \quad (5.6)$$

$$\sum_{i \in P} \sum_{k \in H} \sum_{l \in A} W_{i,j,k,l} = 1 \quad \forall j \in D \quad (5.7)$$

$$\sum_{j \in D} \sum_{k \in H} W_{i,j,k,l} \leq 3 \quad \forall i \in P, \forall l \in A \quad (5.8)$$

$$\sum_{j \in D} \sum_{k \in H} W_{i,j,k,[0]} + W_{i,j,k,[2]} \leq 3 \quad \forall i \in P \quad (5.9)$$

$$\sum_{j \in D} W_{i,j,k,l} \leq 1 \quad \forall i \in P, \forall k \in H, \forall l \in A \quad (5.10)$$

$$\sum_{j \in D} W_{i,j,k,[0]} + W_{i,j,k,[2]} \leq 1 \quad \forall i \in P, \forall k \in H \quad (5.11)$$

$$\sum_{i \in P} \sum_{j \in D} W_{i,j,k,l} \leq S \quad \forall k \in H, \forall l \in A \quad (5.12)$$

$$\sum_{i \in P} \sum_{j \in D} W_{i,j,k,[0]} + W_{i,j,k,[2]} \leq S \quad \forall k \in H \quad (5.13)$$

Como pode-se notar, houve apenas a criação de um conjunto de variáveis binárias que informam se um professor foi alocado para uma determinada disciplina, em um par de dias e um horário. Para esta instância foi considerado os seguintes pares de dias e horários:

$$A = \text{Dias de Aula} \begin{cases} \text{Segunda - Quarta} \\ \text{Terça - Quinta} \\ \text{Quarta - Sexta} \end{cases} \quad H = \text{Horários} \begin{cases} 08:00 - 10:00 \\ 10:00 - 12:00 \\ 13:30 - 15:30 \\ 15:30 - 17:30 \end{cases} \quad (5.14)$$

$$(5.15)$$

A descrição da função objetivo e restrições do modelo são definidas a seguir:

- As restrições (5.5) e (5.6) representam que cada professor pode lecionar no mínimo 2 e no máximo 4 disciplinas;
- A restrição (5.7) representa que cada disciplina deve ser lecionada por somente um professor em um horário e em um dia;
- A restrição (5.8) representa que em um dia, cada professor pode lecionar no máximo 3 disciplinas;
- A restrição (5.9) garante que na quartas-feiras, cada professor lecione no máximo 3 disciplinas;
- A restrição (5.10) representa que cada professor só pode lecionar no máximo uma disciplina, em um par de dias no mesmo horário;
- A restrição (5.11) representa que as disciplinas lecionadas por um professor não podem ser dadas na Segunda-feira|Quarta-feira e Quarta-feira|Sexta-feira no mesmo horário;
- A restrição (5.12) representa que para cada par de dia e horário, o número de professores alocados não pode ser maior que número de salas disponíveis;
- A restrição (5.13) impede que o número de aulas ocorrendo na quarta-feira, extrapole o número de salas.

Note que o modelo considera que Segunda-feira|Quarta-feira e Quarta-feira|Sexta-feira são pares de dias diferentes e que a Quarta-feira está contida nos dois conjuntos, portanto as restrições 5.9 e 5.13 se tornam necessárias, para garantir que a soma de aulas ocorrendo na Quarta-feira não quebrem as restrições 5.8 e 5.12. Por fim, a função objetivo maximiza a afinidade dos professores alocados em cada disciplina, sendo realizada duas punições (constantes p_1 e p_2), a primeira pune soluções as quais sejam alocados mais de 3 disciplinas a um mesmo professor, enquanto a segunda pune alocações onde a soma de aulas no mesmo horário ultrapasse a metade do número de salas disponíveis, buscando evitar que a alocação concentre muitas aulas em um mesmo horário e dia.

5.4 Geração de soluções

O modelo de PLI proposto foi implementado na linguagem de programação Python 3.0 utilizando a biblioteca *MIP* e executado no ambiente de programação em nuvem *Google Colab* em uma máquina virtual com a CPU Intel Xeon 2.20GHz contendo 12,68GB de RAM e

107,72GB de disco para armazenamento. Os resultados obtidos foram armazenados no formato .XLSX informando o código da disciplina, professor alocado, par de dias e horário definido.

Procurando encontrar uma solução condizente com a realidade atual da UFC - Campus Russas, foi realizado 3 testes com a instância criada, diferindo apenas o número de professores disponíveis: 16, 13 e 11. A Figura 5 apresenta o número de disciplinas alocadas para os professores em cada teste realizado, enquanto a Figura 6 apresenta uma parte da solução gerada pelo modelo de PLI considerando 16 professores disponíveis. Esta solução foi a que obteve melhor valor de função objetivo, ou seja, maior afinidade dos professores com as disciplinas alocadas.

Figura 5 – Informações sobre as soluções obtidas.



Fonte: Criado pelo autor.

Figura 6 – Solução obtida com 16 professores, onde estes são representados por letras maiúsculas na primeira coluna.

A	Ética Profissional ES (Ter Qui - 08:00)	Inteligência Artificial CC (Seg Qua - 08:00)	
B	Processos de Software ES (Qua Sex - 13:30)	Engenharia de Software CC (Seg Qua - 08:00)	Qualidade de Software ES (Ter Qui - 10:00)
C	Linguagens Formais e Autômatos (Seg Qua - 08:00)	Matemática Computacional CC (Seg Qua - 10:00)	Compiladores CC (Ter Qui - 13:30)
D	Laboratório de Programação ES (Seg Qua - 10:00)	Laboratório de Programação CC (Seg Qua - 08:00)	Sistemas Operacionais CC/ES (Ter Qui - 08:00)
E	Estruturas de Dados CC (Seg Qua - 08:00)	Lógica para Computação CC (Ter Qui - 08:00)	Lógica para Computação ES (Qua Sex - 15:30)
F	Arquitetura de Computadores ES (Seg Qua - 08:00)	Redes de Computadores CC/ES (Ter Qui - 08:00)	Sistemas Distribuídos CC (Qua Sex - 10:00)
G	Introdução a Processos e Requisitos de Software ES (Ter Qui - 13:30)	Manutenção de Software ES (Seg Qua - 08:00)	Verificação e Validação ES (Ter Qui - 08:00)
H	Introdução à Ciência da Computação CC (Ter Qui - 08:00)	Linguagens de Programação CC/ES (Ter Qui - 13:30)	
I	Ética Profissional CC (Ter Qui - 13:30)	Fundamentos de Banco de Dados CC/ES (Ter Qui - 08:00)	Projeto Detalhado de Software ES (Seg Qua - 08:00)
J	Interação Humano-Computador ES (Ter Qui - 15:30)	Interação Humano-Computador CC (Seg Qua - 08:00)	
K	Fundamentos de Programação ES (Seg Qua - 13:30)	Computação Gráfica CC (Seg Qua - 08:00)	
L	Análise e Projeto de Sistemas ES/CC (Seg Qua - 08:00)	Empreendedorismo ES/CC (Ter Qui - 15:30)	
M	Algoritmos em Grafos ES (Seg Qua - 08:00)	Algoritmos em Grafos CC (Ter Qui - 08:00)	
N	Arquitetura de Computadores CC (Ter Qui - 10:00)	Desenvolvimento de Software para Web CC/ES (Seg Qua - 08:00)	
O	Estruturas de Dados ES (Seg Qua - 08:00)	Estrutura de Dados Avançada CC (Ter Qui - 08:00)	
P	Introdução à Engenharia de Software ES (Seg Qua - 08:00)	Requisitos de Software ES (Ter Qui - 08:00)	

Fonte: Criado pelo autor.

Para o teste considerando 16 professores, foi obtida uma função objetivo de valor igual 149, enquanto para o teste com 13 professores o valor de função objetivo foi 145. Já para o teste utilizando 11 professores, o valor de função objetivo foi 131. É importante mencionar que também foi realizada a computação do valor de função objetivo para a alocação real feita de forma manual pelas coordenações no semestre 2022.2 obtendo-se o valor de 106 pontos, utilizando um total de 14 professores.

6 CONCLUSÕES E TRABALHOS FUTUROS

Este capítulo visa expor as considerações obtidas com a aplicação da metodologia usada para aprimorar a eficiência no solucionamento do problema de alocação de professores a disciplinas na UFC - Campus Russas.

6.1 Considerações gerais

Neste trabalho buscou-se entender as restrições do problema de alocação de professores na UFC - Campus Russas para ser construído um modelo usando os conceitos de Programação Linear Inteira, o qual fosse capaz de melhorar a eficiência da atribuição das disciplinas, focando em maximizar o grau de afinidade do professor com a cadeira que lhe foi destinado.

Ademais, houveram alguns pontos notáveis para a aplicação do modelo criado na instância usada para testes.

- O limite mínimo possível de professores necessários para lecionarem as 44 disciplinas da instância é igual a 11, portanto abaixo desse valor o modelo já não encontra uma solução, uma vez que a restrição 5.6, a qual impede que um professor tenha mais de 4 cadeiras alocadas a ele;
- A remoção de membros no grupo de professores disponíveis, acarretou na diminuição do valor obtido na função objetivo, visto que para suprir a necessidade de cada cadeira estar vinculada a um professor, isso faz com que mais professores precisem lecionar disciplinas as quais eles não possuem tanta afinidade;
- Todas soluções obtidas pelo modelo de PLI apresentam um valor de função objetivo (afinidade dos professores) consideravelmente maior que a alocação real utilizada no semestre 2022.2;
- Um possível ponto que ainda precisa ser melhorado é a prioridade em alocar mais professores em 3 disciplinas. A função objetivo do modelo torna mais favorável a alocação de alguns professores em 2 disciplinas, resultando em um maior número de professores lecionando 4 disciplinas. Em trabalhos futuros, espera-se construir um modelo que favoreça uma alocação mais homogênea.

6.2 Trabalhos Futuros

Em trabalhos futuros, é esperado a realização de mais testes com novas instâncias reais, além de testes com instâncias geradas de maneira randômica com o intuito de avaliar o tempo de execução do modelo para instâncias mais complexas. Tais verificações serão cruciais para avaliar a eficiência do modelo proposto. Além disto, seria interessante compreender melhor algumas requisições específicas da UFC - Campus Russas.

Por fim, também espera-se que em trabalhos futuros, novos modelos sejam criados para que sua solução não aloque apenas os professores às disciplinas, como também os atribua salas de aula, onde as disciplinas deverão ser lecionadas, para tal o modelo deverá levar em consideração fatores como a disponibilidade do e capacidade das sala.

REFERÊNCIAS

- APARECIDO, A. C. **Otimização Combinatória**. 2022. Universidade Estadual de Maringá - Paraná, Disponível em: <https://www.din.uem.br/ademir/pg/otimizacao.html> - Acessado em 5 de julho de 2022.
- BRUSCO, M.; STAHL, S. **Branch-and-Bound Applications in Combinatorial Data Analysis**. Springer New York, 2006. (Statistics and Computing). ISBN 9780387288109. Disponível em: <https://books.google.com.br/books?id=RWo6uz6NcSwC>.
- CARVALHO, J. **Programação Linear: Algoritmos simplex primal, dual, transporte e afetação**. Vida Economica Editorial, 2014. 19-21 p. ISBN 9789897680328. Disponível em: <https://books.google.com.br/books?id=ToWnBAAAQBAJ>.
- FILHO, F. S. d. F. Uma abordagem para o problema de alocação de professores em disciplinas utilizando programação linear inteira. 2016.
- LARA, B. A. S. Um modelo de programação linear para solução do problema da aplicação de recursos em atividades complementares em instituições de ensino superior. **Simpósio Brasileiro de Pesquisa Operacional (SBPO)**, v. 36, p. 381–387, 11 2004.
- PRADO, D. **Programação Linear**. Falconi Editora, 2016. v. 1. (Pesquisa Operacional, v. 1). ISBN 9788598254975. Disponível em: <https://books.google.com.br/books?id=DyfwDwAAQBAJ>.
- SALES, E. d. S. *et al.* Problema de alocação de salas e a otimização dos espaços no centro de tecnologia da ufsm. Universidade Federal de Santa Maria, 2015.
- SILVA, L. F. A. C. *et al.* Modelo de programação linear inteira para o problema de alocação de salas: estudo de caso em uma instituição de ensino superior. Universidade Federal da Paraíba, 8 2019.
- WOLSEY, L. **Integer Programming**. 2. ed. John Wiley Sons, Inc, 2021. 3-5 p. ISBN 9781119606529. Disponível em: <https://books.google.com.br/books?id=knH8DwAAQBAJ>.