

Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação
**Otimização linear inteira para resolução do
problema de formação de equipes: otimização de
múltiplas equipes com mentoria**

Aluno: Gabriel Vinícius dos Santos - NUSP: 11819424
Professora orientadora: Maristela Oliveira dos Santos - SME/ICMC/USP

Relatório final - São Carlos - setembro de 2023

RESUMO: Neste projeto de pesquisa, investigamos meios de resolver de maneira eficiente o problema de formação de equipes, de modo a minimizar os atrasos dos projetos. Isso será alcançado criando equipes eficientes de acordo com as habilidades dos profissionais que as compõem. Dado um conjunto de projetos, cada um exigindo profissionais com habilidades específicas, e dado um conjunto de profissionais, cada um possuindo uma quantidade de habilidades específicas, deve-se selecionar uma equipe ótima de profissionais em termos de maximizar a pontuação total, que é a soma das pontuações de cada projeto. As equipes devem ser formadas com profissionais de modo a satisfazer todas as habilidades requeridas para os projetos. Cada projeto apresenta seu tempo de processamento e um valor de pontuação caso seja concluído antes de seu prazo, e perde um ponto para cada dia posterior ao prazo, até que a pontuação se torne nula. Além disso, os profissionais podem se ajudar com a mentoria, o que permite alocar um profissional com uma unidade de valor de habilidade abaixo do necessário para um projeto. Também para este problema, o valor das habilidades podem aumentar se não forem superiores ao valor necessário, e podem haver diversas equipes com projetos diferentes ao mesmo tempo. Assim, após investigação de trabalhos semelhantes na literatura, desenvolveremos um procedimento heurístico para determinar soluções para o problema.

Palavras-chave: Problema de formação de equipe; Otimização Linear Inteira; Designação; Equipes múltiplas; Mentoria.

1 Introdução

As empresas de diversos setores enfrentam o desafio de formar equipes com profissionais que possuem habilidades e experiências distintas para lidar com diversos projetos. Determinar como compor as equipes de modo que as tarefas sejam executadas da melhor maneira possível, que haja colaboração e transmissão de conhecimento não é uma tarefa fácil. Em diversos casos, há a necessidade de criar equipes de maneira eficiente, de modo a trazer diversos benefícios, como não desperdício de recursos, satisfação dos clientes e economia de tempo. Existem diversos tipos de formação de equipes e muitas aplicações, desde formar o melhor time de algum esporte, dado um conjunto de atletas, até construir diversas equipes de profissionais de tecnologia, sendo que cada um possui conhecimentos específicos, como o domínio de certas linguagens de programação.

Este atual projeto, motivado pela base de dados oferecida pela empresa Google, disponível em GOOGLE (2022), vem por estudar o caso de formação de múltiplas equipes, em que cada equipe se encarrega de um projeto, e que devem ser montadas usando profissionais, também chamados de contribuidores, que possuem habilidades requeridas para os projetos. São considerados um grupo de contribuidores, cada um com suas habilidades, e um grupo de projetos, cada um com seu tempo de execução, sua data de validade, e sua pontuação, dada por um valor que decai para cada dia que o projeto está atrasado, até se tornar nulo. Para este problema, o modelo proposto em Campêlo e Figueiredo (2021) foi usado como abordagem inicial, e devido presença do fator tempo contínuo, também foram estudados os problemas de *flow shop* e escalção de enfermeiros, sendo essenciais para entendimento e construção inicial de modelo matemático do problema, bem como da proposição da abordagem de solução. Desse modo, o problema de formação de equipes busca minimizar os atrasos dos projetos, gerando assim a maior pontuação, e utilizando de heurísticas de otimização para escolher os melhores profissionais para comporem as equipes. Assim, considerando o impacto nas empresas de estudos sobre o problema, as abordagens desenvolvidas neste projeto contribuem com o desenvolvimento tecnológico do país.

2 Definição e modelo matemático para o problema de formação de múltiplas equipes com mentoria

Nesta seção, apresenta-se uma definição do problema, juntamente com ensaios iniciais para a elaboração de um modelo matemático, os quais auxiliam na formulação de estruturas para obter soluções factíveis. Este estudo foi baseado inicialmente nos trabalhos de Campêlo e Figueiredo (2021) e Gutiérrez et al. (2016) que conside-

raram o problema de formação de múltiplas equipes, e não de uma única equipe. Para formulação matemática, índices e parâmetros para este problema, foram também observados os modelos propostos em FLIZICOSKI (2017) e Rahimian et al. (2017), que apresentam a variável tempo em seus problemas. Neste projeto, tempo para alocações de profissionais para as equipes e tempo de processamento dos projetos foram adaptados para o problema de formação de múltiplas equipes. Também Ribeiro et al. (2017) e Juárez et al. (2021) mostraram modelos matemáticos que mais se aproximam do problema deste projeto. Além disso, a mentoria é um novo desafio proposto em GOOGLE (2022), fazendo uma abordagem mais dinâmica para o problema. Sendo assim, considerou-se para este projeto diversas restrições relacionadas ao tempo, junto com a mentoria, que não estavam consideradas nos trabalhos de Campêlo e Figueiredo (2021) e Gutiérrez et al. (2016).

2.1 Modelo matemático

A notação para parâmetros e índices é dada da seguinte forma:

\mathcal{P} sendo um conjunto de projetos os quais os contribuidores são alocados. Os projetos podem ser dados por: $\mathcal{P} = \{p_1, p_2, \dots, p_i\}$.

\mathcal{S} sendo a pontuação de cada projeto em \mathcal{P} se for terminado antes do prazo. Dado por: $\mathcal{S} = \{s_1, s_2, \dots, s_i\}$.

\mathcal{Q} sendo o tempo de processamento de cada projeto em \mathcal{P} . Dado por: $\mathcal{Q} = \{q_1, q_2, \dots, q_i\}$.

\mathcal{D} sendo o prazo cada projeto em \mathcal{P} . Os prazos podem ser dados por: $\mathcal{D} = \{d_1, d_2, \dots, d_i\}$.

\mathcal{C} sendo um conjunto de contribuidores, ou profissionais, que podem ser alocadas em \mathcal{P} . Os contribuidores podem ser dados por $\mathcal{C} = \{c_1, c_2, \dots, c_j\}$.

\mathcal{H} sendo um conjunto de habilidades, pertencentes a cada contribuidor de \mathcal{C} . As habilidades podem ser dadas por $\mathcal{H} = \{h_1, h_2, \dots, h_k\}$.

\mathcal{K}_{jk} sendo o valor da habilidade h_k pertencente a cada contribuidor c_j . Este parâmetro tem valor nulo caso o contribuidor ainda não tenha adquirido a habilidade h_k .

\mathcal{R}_i sendo um conjunto de habilidades requeridas para cada projeto em \mathcal{P} .

As variáveis de decisão são dadas por:

\mathcal{X}_{ij} uma variável binária que indica se o contribuidor c_j é atribuído para o projeto p_i .

\mathcal{Y}_i sendo o dia que o projeto p_i é iniciado, indicando a ordem de início dos projetos.

2.2 Função Objetivo

No modelo atual do problema, visamos obter a maior pontuação de todos os projetos, o que significa minimizar o atraso total de todos os projetos, uma vez que cada dia de atraso, significa uma unidade a menos no valor da pontuação. Sendo Z_i o atraso de cada projeto, temos a seguinte função objetivo:

$$\min \sum_i Z_i$$

Para o problema atual, é possível identificar diversas restrições complexas, como mostrado em Campêlo e Figueiredo (2021), e as principais convergem ao sentido de permitirem que uma equipe seja alocado a um projeto apenas se todas as habilidades requeridas deste projeto foram atendidas, e de não deixar um mesmo contribuidor trabalhar em dois projetos ao mesmo tempo. Como a variável tempo contínuo não está presente nos trabalhos de Campêlo e Figueiredo (2021) e Gutiérrez et al. (2016), e como este projeto pretendia usar modelos matemáticos como base para resolver o problema de formação de múltiplas equipes, e não como objetivo final, optou-se por apresentar posteriormente as restrições relevantes deste problema, em um trabalho futuro. Assim, optou-se pelo desenvolvimento de heurísticas para lidar com o problema.

3 O problema de otimização de múltiplas equipes com mentoria

O problema de formação de equipes com mentoria pode ser compreendido como a tarefa de alocar contribuidores a projetos de maneira a alocar os projetos de maneira mais eficiente, obtendo a melhor pontuação. Nesse cenário, cada projeto é caracterizado por requisitos específicos, como habilidades necessárias, prazos e pontuações associadas. Por exemplo, considerando um projeto que envolva desenvolvimento de software, a equipe alocada deve possuir profissionais com conhecimento em linguagens de programação específicas, e a pontuação do projeto diminui gradualmente após o prazo estabelecido.

O desafio consiste em determinar como formar as equipes de maneira eficaz, levando em consideração as habilidades individuais dos contribuidores e as necessidades de cada projeto. Isso se assemelha ao problema de escalas de trabalho em equipes de enfermagem, onde os membros da equipe devem ser atribuídos a turnos de trabalho e folga de forma otimizada. Assim como no caso das enfermeiras, onde cada turno tem restrições de horários e descanso entre turnos, os contribuidores precisam ser alocados de acordo com suas habilidades e capacidades para

garantir a eficácia da equipe. No entanto, no problema de formação de equipes para projetos, temos uma complexidade devido à interdependência entre os projetos e às restrições de tempo. A alocação inadequada de contribuidores pode levar a atrasos nos projetos, resultando em perda de pontuação.

3.1 Breve Revisão Bibliográfica

Revisão da literatura nos temas *flow shop* (FLIZICOSKI (2017)); escalção de enfermeiros (Rahimian et al. (2017)); e o problema de formação de equipes (Campêlo e Figueiredo (2021), Juárez et al. (2021), Gutiérrez et al. (2016), Ribeiro et al. (2017)), com estudos de diversas propostas de modelagem matemática. Por fim, pesquisa sobre o desenvolvimento de métodos heurísticos principalmente heurísticas construtivas, visando uma heurística para a obtenção de um solução inicial para o problema.

3.2 instâncias da literatura

Na literatura relacionada ao problema de formação de equipes com mentoria, temos seis instâncias, de acordo com o problema apresentando por GOOGLE (2022). Neste contexto, destacam-se as seis instâncias:

1. **Instância "a_an_example"**: Esta instância serve como um exemplo introdutório, com o propósito de ilustrar os conceitos fundamentais. Ela consiste em 3 contribuidores, 3 projetos e 3 habilidades. Embora seja uma instância pequena, ela é valiosa para compreender as dinâmicas básicas da alocação de contribuidores em projetos.
2. **Instância "b_better_start_small"**: A segunda instância apresenta um aumento na escala, contando com 100 projetos, 50 contribuidores e 36 habilidades. Essa instância já oferece um desafio maior ao modelar e otimizar a formação de equipes, considerando um número mais significativo de projetos e contribuidores. A maioria dos testes feitos para estudo do problema foram feitos sobre esta instância.
3. **Instância "c_collaboration"**: Para explorar cenários ainda mais complexos, temos a terceira instância. Aqui, a escala cresce significativamente, com 1000 projetos, 1500 contribuidores e 200 habilidades. Segundo apontado pelo nome da instância, essa simula um ambiente em que a colaboração é um papel crucial.

4. **Instância "d_dense_schedule"**: A quarta instância prioriza a densidade da programação, apresentando 1000 projetos, 500 contribuidores e 100 habilidades. É preciso nesta instância, um maior estudo sobre o problema, uma vez que esta requer vários projetos simultâneos, dificultando a análise algorítmica do problema
5. **Instância "e_exceptional_skill"**: A complexidade aumenta exponencialmente na quinta instância, com 10000 projetos, 800 contribuidores e 800 habilidades. Aqui, o foco recai sobre a gestão de um grande número de projetos e a alocação precisa de contribuidores com habilidades específicas, uma vez que existem mais requisições de habilidades com nível maior pelos projetos.
6. **Instância "f_find_great_mentors"**: A última instância é dotada de ainda mais as abordagens propostas. Ela compreende 19413 projetos, 1000 contribuidores e 500 habilidades. A complexidade dessa instância é dotada pela escolha dos mentores e mentorados corretos, gerando também uma nova complexidade ao problema.

Cada uma dessas instâncias oferece um conjunto único de desafios para testar e aprimorar modelos e algoritmos de formação de equipes com mentoria. Elas refletem cenários diversificados, o que poderia ser encontrado nas empresas e organizações, contribuindo para o desenvolvimento de abordagens diferentes, visando o objetivo do problema.

Ao longo da literatura, de acordo com o problema em GOOGLE (2022), que aborda o problema de formação de equipes com mentoria, temos na Tabela 1 os melhores resultados para as seis instâncias. Esses resultados destacam a eficácia de diversas abordagens e algoritmos na busca por soluções otimizadas. Abaixo estão listados os melhores resultados conhecidos para as seis instâncias consideradas:

Tabela 1: Melhores Resultados da Literatura para as Instâncias

Instância	Melhor Resultado
a - an example	33
b - better start small	1.005.020
c - collaboration	288.508
d - dense schedule	674.945
e - exceptional skill	1.650.488
f - find great mentors	1.194.515

É interessante observar que, ao comparar esses resultados, pode-se notar uma

ampla gama de pontuações alcançadas, indicando a complexidade variável das instâncias e a influência das estratégias de otimização empregadas.

Além desses resultados da literatura, é relevante mencionar o desempenho excepcional do vencedor da competição do Google. O primeiro lugar na competição obteve uma pontuação impressionante de 4.220.236 pontos, demonstrando a excelência das soluções desenvolvidas e a importância contínua da pesquisa nesse campo.

Os resultados acima ressaltam as realizações notáveis alcançadas por meio de diferentes abordagens e algoritmos em problemas de formação de equipes para projetos. A próxima seção apresentará uma metodologia para este problema, seguindo de uma análise comparativa desses resultados, destacando as tendências e implicações relevantes para o avanço dessa área de estudo.

4 Abordagem da solução

A abordagem para resolver o problema de formação de equipes eficientes baseou-se na implementação de um código em Python 3.9.12. A lógica principal do código envolve uma estratégia de ordenação criteriosa dos projetos, contribuidores e habilidades, criando uma base sólida para a alocação otimizada.

Dentro do código, aplicou-se diversas heurísticas construtivas para obtenção de uma solução inicial, que permitiram estabelecer cenários iniciais para as alocações. Essas heurísticas têm como objetivo encontrar configurações promissoras para as equipes de contribuidores, considerando as restrições de habilidades e prazos dos projetos.

Uma das principais etapas da lógica é a alocação propriamente dita, que busca combinar os projetos com os contribuidores adequados, levando em conta as ordenações previamente estabelecidas. Ao seguir essa abordagem, busca-se maximizar a eficiência das equipes e minimizar possíveis atrasos nos projetos.

Um aspecto importante da atual abordagem foi a repetição do processo para cada instância, a fim de obter resultados robustos e comparativos. Cada instância foi executada três vezes para coletar os resultados e permitir uma análise comparativa das heurísticas de solução inicial e das trocas realizadas.

Com a combinação de ordenação criteriosa, heurísticas de solução inicial e repetição das instâncias, buscou-se explorar diferentes abordagens e cenários, contribuindo para a obtenção de soluções mais eficazes para o problema de formação de equipes eficientes em projetos.

4.1 Heurísticas de solução inicial

Ao abordar o problema de formação de equipes eficientes para projetos, utilizou-se de uma série de heurísticas de solução inicial para criar cenários de alocação promissores. Essas heurísticas desempenham um papel crucial na busca por soluções otimizadas, estabelecendo as bases para a formação das equipes.

As regras de ordenação de projetos que foram empregadas como heurísticas de solução inicial incluem:

1. **Shortest Processing Time (SPT)**: Esta heurística prioriza projetos com o menor tempo de processamento. Ela busca minimizar o tempo necessário para concluir os projetos, otimizando a utilização dos contribuidores.
2. **Longest Processing Time (LPT)**: A heurística LPT seleciona projetos com o maior tempo de processamento. Isso pode ser benéfico em cenários onde a complexidade do projeto exige uma alocação mais adequada de recursos.
3. **Earliest Due Date (EDD)**: A heurística EDD atribui prioridade aos projetos com a data de entrega mais próxima. Ela visa minimizar atrasos nos projetos, garantindo que os prazos sejam cumpridos de forma eficiente.
4. **Minimum Slack Time (MST)**: Essa heurística considera o tempo disponível até a data de entrega. Projetos com menor margem de tempo têm prioridade, garantindo que os projetos mais urgentes sejam atendidos.
5. **Critical Ratio (CR)**: A heurística CR calcula a relação entre o prazo e o tempo de processamento de cada projeto. Projetos com CR mais alto são selecionados primeiro, visando otimizar a relação entre tempo e valor.
6. **SLACK (SLK)**: Esta heurística leva em consideração o slack time (tempo de folga) disponível para cada projeto. Projetos com menos tempo de slack são priorizados para garantir uma alocação mais eficiente dos contribuidores.
7. **MANPOWER (MPW)**: A heurística MANPOWER calcula o valor da pontuação total do projeto dividido pelo número de contribuidores alocados. Isso busca otimizar a utilização dos recursos humanos disponíveis.

É interessante observar que, na literatura, as heurísticas EDD e SPT são frequentemente reconhecidas como as mais eficazes para problemas de agendamento ("scheduling"). Essas heurísticas têm se mostrado valiosas para a formação de equipes eficientes, permitindo uma alocação inteligente de recursos e otimização dos resultados.

As heurísticas de solução inicial desempenham um papel fundamental no desenvolvimento de abordagens robustas para o problema de formação de equipes. A próxima seção apresentará a aplicação dessas heurísticas às instâncias consideradas, detalhando os resultados obtidos e as implicações para a otimização das equipes de contribuidores.

Na busca por formação de equipes eficientes em problemas de projetos, a escolha adequada dos contribuidores desempenha um papel crucial na obtenção de resultados otimizados. No âmbito desse desafio, utilizou três heurísticas de escolha de contribuidores, cada uma visando maximizar a eficiência da equipe formada.

As heurísticas de escolha de contribuidores utilizadas incluem:

1. **First to Find (Ftf)**: A heurística "First to Find" prioriza a escolha do primeiro contribuidor que atende aos requisitos do projeto e possibilita melhorias. Isso permite que sejam identificadas oportunidades de aprimoramento e otimização da equipe desde o início da alocação.
2. **Nearest (Nst)**: A heurística "Nearest" seleciona o contribuidor cujo nível de habilidade está mais próximo do valor requerido pelo projeto. Essa abordagem visa otimizar a alocação, escolhendo indivíduos cujas habilidades se alinham de forma mais eficaz com as necessidades do projeto.
3. **Farest (Fst)**: A heurística "Farest" prioriza o contribuidor cujo nível de habilidade está mais distante do valor requerido pelo projeto. Isso pode ser benéfico para projetos que demandam especializações específicas, maximizando a eficácia da equipe.

É importante destacar que, em todos os casos, as heurísticas de escolha de contribuidores são projetadas para priorizar contribuidores que oferecem oportunidades de aprimoramento. O foco inicial é atribuído aos indivíduos que podem se beneficiar de um processo de aprimoramento de suas habilidades, enriquecendo a equipe ao longo do tempo.

Além disso, as heurísticas buscam aproveitar as possibilidades de mentoreamento. Contribuidores com habilidades próximas às necessidades do projeto têm a chance de atuar como mentores, auxiliando na formação e no desenvolvimento de outros membros da equipe. Isso promove um aumento geral nas habilidades e na eficiência da equipe, maximizando os ganhos de valor.

As heurísticas de escolha de contribuidores desempenham um papel essencial no processo de alocação e formação de equipes eficientes. A seção subsequente apresenta os resultados obtidos pela aplicação dessas heurísticas às instâncias consideradas, destacando como elas contribuíram para a otimização dos projetos e dos recursos humanos.

4.2 Abordagem Proposta

O algoritmo de solução inicial é uma parte fundamental deste trabalho, pois estabelece uma base a partir da qual a busca por soluções ótimas a serem exploradas e entendidas. O pseudocódigo a seguir descreve a estrutura geral do algoritmo de solução inicial:

Algorithm 1 Algoritmo de Solução Inicial

```
1: procedure SOLUCAOINICIAL(Projetos, Contribuidores)
2:    $Projetos \leftarrow \text{OrdenaçãoInicial}(Projetos)$ 
3:   for projeto em  $Projetos$  do
4:      $equipe \leftarrow \text{CriarEquipeVazia}(projeto)$ 
5:     for contribuidor em  $Contribuidores$  do
6:       if AlocaMentorado(projeto, contribuidor, equipe) then
7:         AloqueContribuidor(projeto, contribuidor, equipe)
8:       else
9:         if AlocaPerfeito(projeto, contribuidor, equipe) then
10:          AloqueContribuidor(projeto, contribuidor, equipe)
11:        else
12:          if AlocaNormal(projeto, contribuidor, equipe) then
13:            AloqueContribuidor(projeto, contribuidor, equipe)
14:          end if
15:        end if
16:      end if
17:    end for
18:     $equipe \leftarrow \text{EscolhaDeContribuidores}(equipe)$ 
19:    AdicioneEquipe(Equipes, equipe)
20:  end for
21:  return Equipes
22: end procedure
```

Sendo assim, o algoritmo começa com listas de projetos e contribuidores e, em seguida, itera sobre os projetos, criando equipes vazias para cada um e alocando contribuidores de acordo com as restrições do problema. Uma série de condicionais avalia o quanto bom seria alocar o contribuidor. Primeiro, observamos se ele pode ser considerado um mentorado, depois se este contribuidor possui o nível de exigência de habilidade suficiente para aumentar seu próprio nível. Sendo assim, o algoritmo busca uma escolha de contribuidor. Uma função no início do código escolhe a heurística de ordenação de projetos. Por fim, depois de avaliados todos os contribuidores, uma heurística de escolha de contribuidores avalia quais os me-

lhores para a equipe do projeto atual. Se a heurística de escolha de contribuidores **Ftf** foi escolhida, então não ocorre essa última etapa, pois os contribuidores são os primeiros a serem atribuídos.

Ao estudar e trabalhar com diferentes heurísticas, o algoritmo de solução inicial permite uma melhor compreensão do problema. Ele foi projetado para ser escalável, permitindo sua aplicação a casos de teste com diferentes níveis de complexidade. As combinações de heurísticas mais eficazes foram identificadas e testadas em casos mais complexos, comparando-se com as melhores pontuações conhecidas até o momento.

4.3 Heurísticas para troca de contribuidores

O pseudo-código abaixo mostra o algoritmo usado para encontrar melhores soluções, inspirado nos algoritmos de perturbação local presentes em outros trabalhos (LAJÇI; MISINI, 2023). Essa abordagem consiste em uma heurística de melhoria, que tenta, com a troca, melhorar a qualidade da solução.

Algorithm 2 Troca de Contribuidores para Melhorar a Solução

```

1: procedure TROCARCONTRIBUIDORES(SolucaoInicial, Iteracoes)
2:   MelhorSolucao  $\leftarrow$  SolucaoInicial
3:   for  $i \leftarrow 1$  to Iteracoes do
4:     ContribuidorA  $\leftarrow$  SelecionarContribuidores(SolucaoInicial)
5:     ContribuidorB  $\leftarrow$  SelecionarContribuidores(SolucaoInicial)
6:     SolucaoTemporaria  $\leftarrow$  CopiarSolucao(SolucaoInicial)
7:     RealizarTroca(SolucaoTemporaria, ContribuidorA, ContribuidorB)
8:     if Pontuacao(SolucaoTemporaria) > Pontuacao(MelhorSolucao) then
9:       MelhorSolucao  $\leftarrow$  SolucaoTemporaria
10:    end if
11:  end for
12:  return MelhorSolucao
13: end procedure

```

Sendo assim, o objetivo deste algoritmo é fazer uma perturbação na solução, obtendo uma nova solução diferente da inicial. O algoritmo usa da aleatoriedade para selecionar dois contribuidores, sendo que um deles deve estar presente em algum dos projetos. Após selecionados, a troca dos contribuidores é feita e estimasse uma nova pontuação de acordo com a solução nova encontrada. No final do algoritmo, temos a melhor solução para um número fixo de iterações.

5 Resultados

Neste projeto, os resultados são demonstrados na Tabela 2, que exibe os resultados mais destacados alcançados nas seis instâncias disponíveis, conforme citado por GOOGLE (2022). A linha "total" ilustra o valor acumulado mais elevado, que foi alcançado por um único algoritmo, enquanto cada instância individual no estado da arte obteve seus valores por meio de algoritmos distintos.

Tabela 2: Instâncias, Soluções do Estado da arte e soluções encontradas neste trabalho.

Inst.	Estado da arte	Solução - Projeto
a	33	33
b	1.005.020	855.630
c	288.508	199.049
d	674.945	173.626
e	1.650.488	1.610.072
f	1.194.515	543.471
total	4.220.236	3.381.881

Na Tabela 3, são apresentadas métricas de comparação de resultados, especificamente o GAP, para cada instância. Essa métrica avalia o quão distante a solução encontrada está em relação ao Estado da Arte. Além disso, a tabela fornece informações sobre quais heurísticas foram empregadas para obter as soluções encontradas. A métrica GAP pode ser definido como:

$$GAP = \frac{\text{MelhorSolução} - \text{SoluçãoProposta}}{\text{MelhorSolução}} 100\% \quad (1)$$

Tabela 3: Instâncias , GAP das soluções, Heurísticas usadas nas melhores soluções para cada instância

Instância	GAP	Heurística 1	Heurística 2
a	0%	SPT	Ftf
b	14,86%	MST	Ftf
c	31,00%	LPT	Ftf
d	74,28%	SPT	Ftf
e	24,49%	CR	Ftf
f	54,50%	SLK	Ftf
Total	19,87%	-	Ftf

Por fim, na Tabela 4, são apresentados os resultados do algoritmo de melhoria local em relação às instâncias correspondentes. Adicionalmente, o GAP' é uma métrica utilizada para avaliar o grau de melhoria alcançado na solução. A métrica GAP' pode ser definido como:

$$\text{GAP}' = \frac{\text{NovaSolução} - \text{PrimeiraSolução}}{\text{NovaSolução}} 100\% \quad (2)$$

Tabela 4: Instâncias , Soluções encontradas e Melhora das soluções (GAP')

Teste	1ª Sol.	2ª Sol.	GAP'
b - SPT	800981	800981	0%
b - LPT	800975	800975	0%
b - EDD	800918	769399	3,94%
b - MST	800991	769472	3,94%
b - CR	800991	800991	0%
b - SLK	800930	769411	3,94%
b - MPW	800989	800989	0%

5.1 Análise da solução obtida

Uma vez com os dados obtidos, podemos fazer uma análise fundamental das soluções, garantindo conclusões importantes sobre como este problema funciona. Primeiramente, Figura 1 mostra os resultados da instância "a". Nela observamos o problema como um exemplo, e o algoritmo performou muito bem, em que algumas heurísticas de ordenação apenas deixaram diminuir 3 pontos da melhor solução.

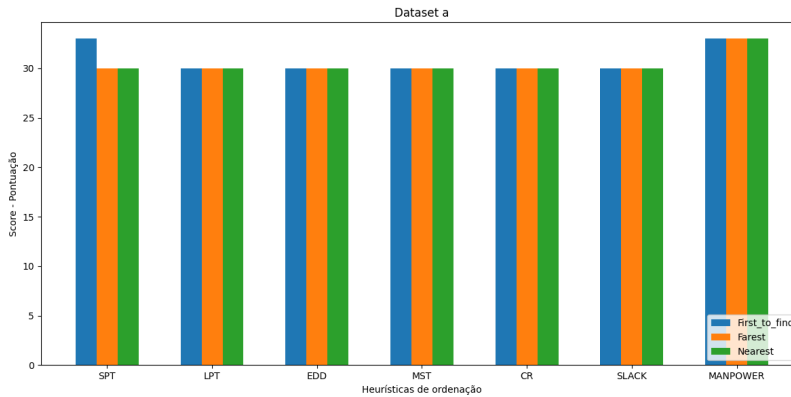


Figura 1: Resultados por heurística de solução inicial

Observando o dataset "b", temos algumas conclusões interessantes. Primeiramente observa-se que as heurísticas de ordenação tiveram resultados muito parecidos. Além disso, temos que a heurística de escolha de colaboradores "first_to_find" apresenta os melhores valores para todas esta instância. Assim como no primeiro dataset, o tempo de execução desta instância pode ser considerando mínimo, tendo a resposta em poucos segundos.

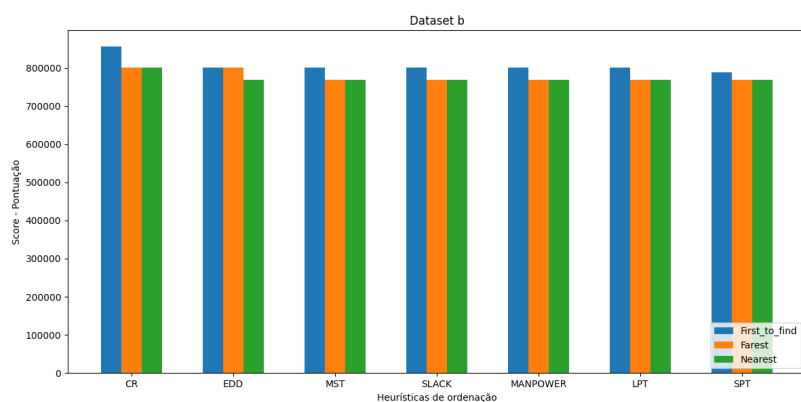


Figura 2: Resultados por heurística de solução inicial

Temos agora o dataset "c". Neste observamos ainda a perpetuação da heurística de escolha de colaboradores "first_to_find" como uma boa escolha. Além disso, a heurística "nearest" também apresentou bons resultados. Neste dataset, as regras de ordenação de projetos SPT e LPT foram as melhores, indicando um dataset que precisa de gerencia melhor dos projetos. O tempo médio de execução do algoritmo ficou em 5 horas.

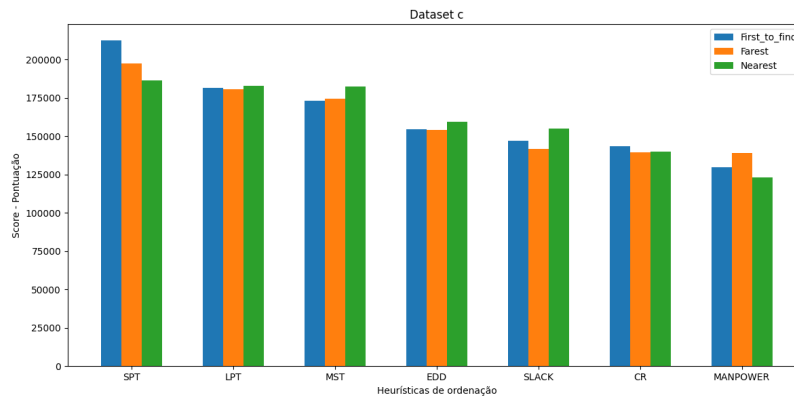


Figura 3: Resultados por heurística de solução inicial

Já agora o dataset "d" aconteceu algo inesperado, em que a solução converge para um valor fixo, não importa quais heurísticas escolhamos. Sendo assim, conclui-se que para este dataset, apenas um algoritmo que busca ordenar e alocar as equipes dos projetos não foi o ideal, sendo um dataset fora do dados esperados. O tempo de execução ficou em 45 minutos.

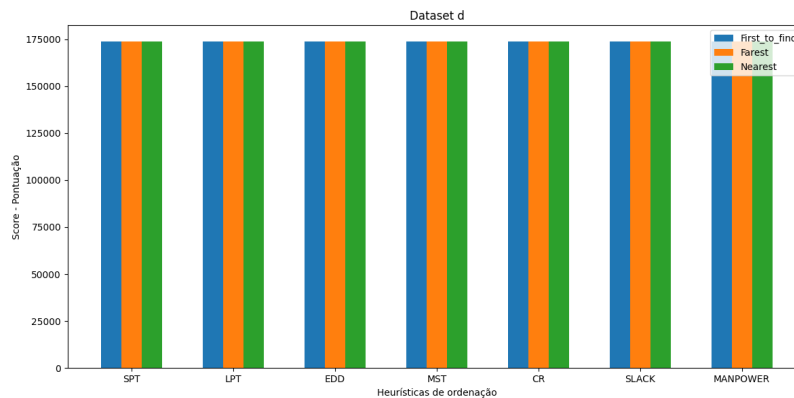


Figura 4: Resultados por heurística de solução inicial

Para as duas ultimas instâncias, optou-se por apenas escolher a heurística que tivesse a melhor performance de acordo com os datasets anteriores. Sendo assim, a heurística de escolha de colaboradores "first_to_find" foi a selecionada, uma vez que apresentou bons resultados e seu desempenho algoritmo diminui o tempo de execução. De acordo com a Figura 5, observamos que as heurísticas "CR", "SLACK",

"MST" e "EDD" tiveram um desempenho melhor. O tempo de execução médio foi de 20 horas.

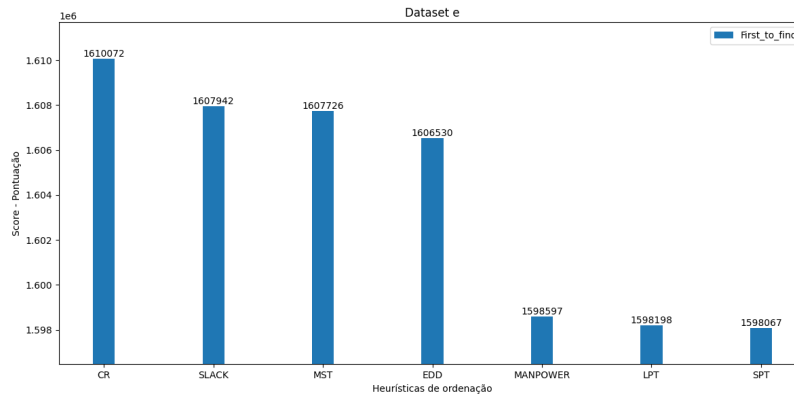


Figura 5: Resultados por heurística de solução inicial

Por fim, a Figura 6 mostra os resultados do dataset "f". Neste observamos que as heurísticas "SLACK", "EDD", "CR" e "SPT" tiveram um desempenho melhor. O tempo de execução médio foi de 17 horas. As instâncias, bem como o código-fonte do algoritmo utilizado neste trabalho pode ser encontrado em:

[https://github.com/GVS2001/TPM_Teamwork-with-Mentoring-Problem].

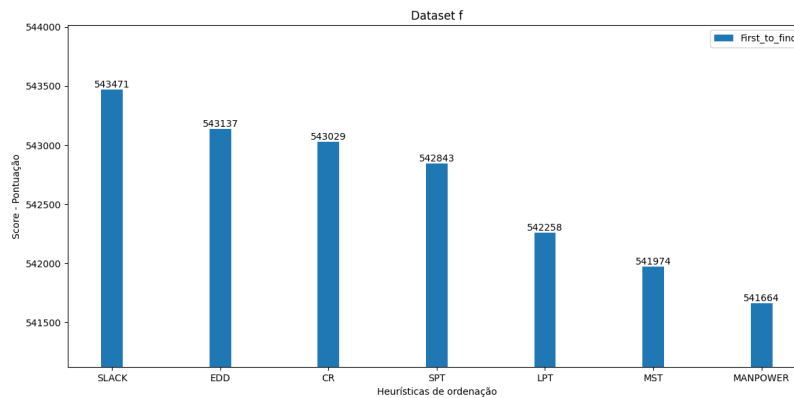


Figura 6: Resultados por heurística de solução inicial

5.2 Considerações sobre as Heurísticas

As heurísticas de ordenação de projetos não tiveram impacto significativo nos resultados, enquanto a heurística "Primeiro a encontrar (Ftf)", para a seleção de contribuidores desempenhou um papel crucial na obtenção de boas soluções iniciais. A análise também mostrou que as características das instâncias, como o número de contribuidores, projetos e habilidades, influenciaram o desempenho do algoritmo. Apenas uma instância ("d") apresentou resultados divergentes, e sua diferença foi considerável em relação ao melhor resultado conhecido, como também observado em outros estudos semelhantes (LAJÇI; MISINI, 2023). Uma análise temporal mostra que o algoritmo não se torna muito escalável, e para grandes instâncias, pode demorar tempos consideráveis.

A busca por soluções mais avançadas para abordar a complexidade da Otimização de Múltiplas Equipes com Mentoria abre um horizonte repleto de oportunidades instigantes para investigações futuras. A abordagem feita neste estudo visa desenvolver soluções iniciais eficazes e compreender o problema de forma eficiente. No entanto, é essencial explorar outras estratégias, como ilustrado em Lajçi e Misini (2023). Além disso, direcionar novas abordagens, examinando o impacto de algoritmos que otimizem a alocação de colaboradores, conforme apresentado em Campêlo e Figueiredo (2021), pode resultar em aprimoramentos substanciais nas soluções iniciais e, conseqüentemente, em resultados superiores. A incorporação de abordagens de aprendizado de máquina e a utilização de modelos de otimização para a seleção de colaboradores têm o potencial de prever tendências e cenários futuros com base nas instâncias existentes, agregando valor prático significativo às investigações futuras neste problema. Essas direções de pesquisa prometem enriquecer consideravelmente nossa compreensão do problema e suas implicações práticas, como em ambientes empresariais.

6 Conclusão

Neste estudo, investigamos a construção de algoritmos de solução inicial para resolver o problema de Otimização de Múltiplas Equipes com Mentoria. Os resultados obtidos foram comparados com o Estado da Arte, e comparados com métricas para avaliar o quanto chegamos perto da melhor solução. Foram usados sete heurísticas de ordenação de projetos e três heurísticas de escolha de contribuidores, buscando estudar o problema e gerar as melhores soluções iniciais possíveis. As heurísticas de escolha de contribuidores mostraram ser eficazes em casos específicos, e cada dataset presente neste trabalho teve suas melhores soluções com heurísticas diferentes. Já para as heurísticas de escolha de contribuidores, houve resultados melhores quando presentes na heurística "First to find (Ftf)", que esco-

lhe o primeiro contribuidor encontrado que gera uma boa pontuação para a equipe formada. Foi também demonstrado que características das instâncias do presente problema, como o número de colaboradores, projetos e habilidades, influenciaram o desempenho dos algoritmos, como mostrado pelo aumento do tempo de execução do algoritmo, além da quarta instância que apresentou convergência para quaisquer heurísticas usadas. O algoritmo de perturbação local (busca local) presente neste trabalho não demonstrou desempenho eficaz, mas seu estudo foi de alta relevância. Neste contexto, é pertinente ressaltar a significância das heurísticas empregadas na concepção de soluções iniciais de alta qualidade, visando à obtenção de soluções ótimas. A utilização de abordagens baseadas em heurísticas para a geração da solução inicial revelou-se substancialmente muito eficaz, sendo uma boa abordagem para um problema de otimização com um grande espaço de soluções possíveis. A solução inicial desempenha um papel central como ponto de partida fundamental no processo de otimização, sendo que sua formulação criteriosa pode exercer um impacto considerável na qualidade global da solução resultante. Resumidamente, este estudo oferece uma contribuição significativa para o aprofundamento da compreensão das heurísticas relacionadas à geração de soluções iniciais na abordagem de problemas de otimização em contextos do mundo real, exemplificado pelo cenário da Otimização de Múltiplas Equipes com Mentoria. Os estudos e as perspectivas adquiridas neste trabalho constituem um recurso valioso para o avanço na concepção de algoritmos de otimização eficazes aplicáveis a domínios de problemas parecidos, o que, por sua vez, aprimora substancialmente a capacidade de tomada de decisões e alocação eficiente de recursos para diversos cenários de problemas envolvendo múltiplas equipes e problemas de alocação semelhantes.

Agradecimentos: Gostaria de expressar meus sinceros agradecimentos a todos aqueles que contribuíram de forma direta e indireta para a conclusão deste estudo. Primeiramente e mais importante, a professora orientadora Maristela Oliveira dos Santos, cuja orientação, conhecimento e apoio foram fundamentais para o desenvolvimento deste trabalho. Também estendo minha gratidão aos amigos e colaboradores que compartilharam ideias valiosas e feedback construtivo ao longo desta jornada de pesquisa. Além disso, reconhece o apoio das Instituições que deram vida a este projeto, como o Conselho Nacional de Desenvolvimento Científico e Tecnológico - Cnpq, o Instituto de Ciências Matemáticas e de Computação - ICMC, e a Universidade de São Paulo - USP. Por fim, gostaríamos de agradecer a minha família e amigos pelo constante incentivo e apoio emocional. Este estudo, que começou como um sonho meu de resolver o presente problema, não teria sido possível sem o apoio e a colaboração de todos vocês. Muito obrigado.

Referências Bibliográficas

CAMPÊLO, M.; FIGUEIREDO, T. F. Integer programming approaches to the multiple team formation problem. *Computers & Operations Research*, v. 133, p. 105354, 2021. ISSN 0305-0548. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0305054821001325>>.

FLIZICOSKI, A. L. V. Aplicação da minimização do atraso total em ambiente de máquina única com tempos de setup dependentes da sequência. 2017. Disponível em: <<http://repositorio.utfpr.edu.br/jspui/handle/1/16135>>.

GOOGLE. *Mentorship and Teamwork - Data*. fev. 2022. Disponível em: <<https://codingcompetitions.withgoogle.com/hashcode/round/00000000008caae7/000000000098afc8>>.

GUTIÉRREZ, J. H.; ASTUDILLO, C. A.; BALLESTEROS-PÉREZ, P.; MORA-MELIÀ, D.; CANDIA-VÉJAR, A. The multiple team formation problem using sociometry. *Computers & Operations Research*, v. 75, p. 150–162, 2016. ISSN 0305-0548. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0305054816301198>>.

JUÁREZ, J.; SANTOS, C. P.; BRIZUELA, C. A. A comprehensive review and a taxonomy proposal of team formation problems. *ACM Comput. Surv.*, Association for Computing Machinery, New York, NY, USA, v. 54, n. 7, jul 2021. ISSN 0360-0300. Disponível em: <<https://doi.org/10.1145/3465399>>.

LAJÇI, U.; MISINI, E. Exploring local search metaheuristics for optimizing the mentorship and teamwork problem. 05 2023.

RAHIMIAN, E.; AKARTUNALI, K.; LEVINE, J. A hybrid integer programming and variable neighbourhood search algorithm to solve nurse rostering problems. *European Journal of Operational Research*, v. 258, n. 2, p. 411–423, 2017. ISSN 0377-2217. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0377221716307822>>.

RIBEIRO, J.; SIQUITELLI, C.; LEONETI, A.; COSTA, A. Um método e um programa para a formação e logística de equipes de trabalho. 2017. Disponível em: <<https://www.proquest.com/scholarly-journals/um-metodo-e-programa-para-formacao-logistica-de/docview/2110083102/se-2>>.