

Otimização Multiobjetivo e Tomada de Decisão para o Problema de Atribuição de Tarefas a Agentes

Gabriel Rolla Ferreira

Universidade Federal de Minas Gerais

Matrícula: 2022038457

Lucas Frazão Moreira

Universidade Federal de Minas Gerais

Matrícula: 2020056857

Lucas Pimenta Braga

Universidade Federal de Minas Gerais

Matrícula: 2023034552

Mateus de Souza Gontijo

Universidade Federal de Minas Gerais

Matrícula: 2020053530

Abstract— Este trabalho aborda o Problema de Atribuição Generalizada sob uma ótica multiobjetivo, visando minimizar simultaneamente o custo total de execução e o desequilíbrio de carga entre os agentes. A metodologia proposta utiliza a metaheurística *Variable Neighborhood Search* (VNS) combinada com uma construção inicial GRASP. Foram implementadas e comparadas duas abordagens de escalarização: Soma Ponderada e o Método ϵ -Restrito. Os resultados demonstram a superioridade do método ϵ -Restrito na geração de uma fronteira de Pareto bem distribuída. Por fim, aplicou-se uma etapa de tomada de decisão multicritério (TOPSIS e Agregação Ponderada) considerando métricas de robustez, resultando na seleção de soluções de compromisso adequadas a diferentes perfis de decisores.

I. INTRODUÇÃO

Atribuição eficiente de tarefas a agentes é um problema clássico de otimização que aparece em diversos contextos práticos, como a distribuição de atividades entre trabalhadores, o escalonamento de processos em máquinas, ou a alocação de recursos em sistemas computacionais. Em todos esses cenários, a má distribuição pode gerar custos elevados ou sobrecarga de determinados agentes, comprometendo o desempenho global do sistema.

Neste trabalho, considera-se uma instância com 5 agentes e 50 tarefas, em que cada tarefa possui um custo específico de execução associado a cada agente e uma demanda de recursos que deve ser atendida respeitando a capacidade de cada agente. Assim, é necessário decidir a alocação de todas as tarefas de forma a garantir que nenhuma capacidade seja excedida e que cada tarefa seja executada por exatamente um agente.

O problema é tratado de forma multiobjetivo, com duas funções principais em análise: (i) a minimização do custo total de execução e (ii) a minimização do desequilíbrio na distribuição de carga entre agentes. A primeira etapa do trabalho concentrou-se nas versões mono-objetivo, de forma a estabelecer valores de referência (ótimos escalarizados) e validar o algoritmo de busca. A partir disso, este relatório apresenta os resultados da otimização multiobjetivo e da etapa de tomada de decisão multicritério.

II. TRABALHOS RELACIONADOS

O Problema de Atribuição Generalizada (GAP) pertence à classe de problemas NP-difícéis, exigindo métodos heurísticos

eficientes para a resolução de instâncias de grande porte. No contexto de otimização combinatória, metaheurísticas baseadas em busca local e construção gulosa aleatorizada têm se destacado.

Hansen e Mladenović [1] formalizaram a *Variable Neighborhood Search* (VNS), demonstrando sua eficácia ao explorar sistematicamente múltiplas vizinhanças para escapar de ótimos locais. Complementarmente, Mateus et al. [2] aplicaram com sucesso o GRASP (*Greedy Randomized Adaptive Search Procedure*) em variações complexas do problema de atribuição, validando o uso de construções semigulosas.

No âmbito da otimização multiobjetivo, onde os objetivos são frequentemente conflitantes, a literatura foca na obtenção de um conjunto de soluções não-dominadas (Fronteira de Pareto) [3]. Técnicas de escalarização, como a Soma Ponderada e o Método ϵ -Restrito, são fundamentais para transformar problemas vetoriais em escalares, permitindo o uso de algoritmos mono-objetivo [4]. Para a etapa final de seleção, métodos de auxílio à decisão multicritério, como o TOPSIS proposto por Hwang e Yoon [5], oferecem uma abordagem estruturada baseada na distância à solução ideal.

III. DEFINIÇÃO DO PROBLEMA

O problema de atribuição de tarefas a agentes consiste em designar um conjunto de tarefas \mathcal{T} , de cardinalidade n , a um conjunto de agentes \mathcal{A} , de cardinalidade m , de modo a respeitar restrições de capacidade e otimizar critérios de custo e equilíbrio.

Para a instância considerada neste trabalho, são fornecidos 5 agentes ($m = 5$) e 50 tarefas ($n = 50$). As informações necessárias (parâmetros) estão disponíveis em arquivos fornecidos pelo professor, contendo:

- uma matriz de custos c_{ij} , que representa o custo de atribuir a tarefa $j \in \mathcal{T}$ ao agente $i \in \mathcal{A}$;
- uma matriz de recursos a_{ij} , que representa a quantidade de recursos consumidos pelo agente $i \in \mathcal{A}$ ao executar a tarefa $j \in \mathcal{T}$;
- um vetor de capacidades b_i , que indica o total de recursos disponíveis para cada agente $i \in \mathcal{A}$.

Variáveis de decisão

Define-se uma variável de decisão binária x_{ij} :

$$x_{ij} = \begin{cases} 1, & \text{se a tarefa } j \in \mathcal{T} \text{ for atribuída ao agente } i \in \mathcal{A}, \\ 0, & \text{caso contrário.} \end{cases} \quad (1)$$

Restrições

O modelo deve obedecer às seguintes restrições:

- 1) **Atribuição única das tarefas:** cada tarefa deve ser realizada exatamente por um agente:

$$\sum_{i=1}^m x_{ij} = 1, \quad \forall j \in \mathcal{T}. \quad (2)$$

- 2) **Limite de capacidade dos agentes:** o consumo de recursos total das tarefas atribuídas a um agente não pode ultrapassar sua capacidade:

$$\sum_{j=1}^n a_{ij} \cdot x_{ij} \leq b_i, \quad \forall i \in \mathcal{A}. \quad (3)$$

- 3) **Domínio das variáveis:** as variáveis de decisão são binárias:

$$x_{ij} \in \{0, 1\}, \quad \forall i \in \mathcal{A}, \forall j \in \mathcal{T}. \quad (4)$$

IV. FORMULAÇÃO MATEMÁTICA

O problema é representado por meio de duas funções objetivo conflitantes, além das restrições já apresentadas.

A. Função objetivo f_1 : minimização do custo total

A primeira função, $f_1(\cdot)$, busca reduzir o custo global de execução de todas as tarefas:

$$f_1(x) = \sum_{i=1}^m \sum_{j=1}^n c_{ij} \cdot x_{ij}, \quad (5)$$

em que c_{ij} representa o custo de atribuir a tarefa j ao agente i .

B. Função objetivo f_2 : minimização do desequilíbrio de carga

A segunda função, $f_2(\cdot)$, busca equilibrar a distribuição das tarefas entre os agentes, reduzindo a diferença entre os agentes mais e menos sobrecarregados em termos de recursos utilizados:

$$f_2(x) = \max_{i \in \mathcal{A}} \left(\sum_{j=1}^n a_{ij} \cdot x_{ij} \right) - \min_{i \in \mathcal{A}} \left(\sum_{j=1}^n a_{ij} \cdot x_{ij} \right). \quad (6)$$

C. Problema multiobjetivo

Assim, o problema de atribuição de tarefas a agentes é formulado como um problema de otimização multiobjetivo:

$$\text{Minimizar} \quad (f_1(x), f_2(x))$$

sujeito às restrições de atribuição, capacidade e domínio das variáveis.

V. ALGORITMO DE SOLUÇÃO

Foi implementada uma variação da metaheurística **Variable Neighborhood Search (VNS)** [1]. O mesmo algoritmo foi utilizado nas três etapas: otimização mono-objetivo (para obter valores de referência), escalarização multiobjetivo (para gerar a fronteira de Pareto) e geração das alternativas usadas na tomada de decisão multicritério.

A. Modelagem computacional

Uma solução é representada por um vetor de tamanho n , em que a posição j indica qual agente executa a tarefa j . Essa codificação é naturalmente factível em relação à restrição de atribuição única; a viabilidade em relação às capacidades é controlada na construção inicial e na avaliação das vizinhanças.

B. Estruturas de vizinhança

Foram usadas três estruturas de vizinhança:

- 1) **Realocação (shift):** move uma tarefa de um agente para outro.
- 2) **Troca simples (swap):** troca duas tarefas entre dois agentes.
- 3) **Dupla troca (2-swap):** troca simultaneamente duas tarefas de agentes distintos.

Na busca local, a vizinhança principal utilizada foi a de **realocação (shift)**, pela boa relação entre tamanho da vizinhança e ganho médio observado.

C. Heurística construtiva (GRASP)

Para gerar a solução inicial, foi implementada uma heurística construtiva baseada no princípio **GRASP** (Greedy Randomized Adaptive Search Procedure) [2].

Para cada tarefa, é construída uma *lista restrita de candidatos* (RCL) contendo os agentes viáveis cujo custo está dentro de um limiar α do melhor custo observado para aquela tarefa. Neste trabalho, utilizou-se $\alpha = 0,3$. Um agente é então selecionado aleatoriamente dessa lista, desde que a atribuição não viole a capacidade correspondente.

Isso garante que cada execução do VNS parte de um ponto inicial diferente, mas ainda assim de qualidade razoável, aumentando a exploração do espaço de busca.

D. Estratégia de refinamento (busca local)

Foi utilizada a busca local do tipo **Best Improvement**. Dada uma solução, o algoritmo explora exaustivamente *toda* a vizinhança de **realocação (shift)**.

Para cada uma das $n = 50$ tarefas, o algoritmo testa movê-la para cada um dos $m - 1 = 4$ outros agentes (totalizando aproximadamente 200 movimentos possíveis). A melhoria que trouxe o maior ganho na função objetivo escalar (dependendo da etapa: f_1 , f_2 ou alguma combinação escalarizada) é então aplicada. O processo se repete até que nenhum movimento de *shift* seja capaz de melhorar a solução, garantindo que um ótimo local (para esta vizinhança) seja atingido.

E. Esboço do VNS utilizado

De forma resumida, o VNS adotado pode ser descrito como:

- 1) Gerar solução inicial x por GRASP.
- 2) Definir a lista de vizinhanças \mathcal{N}_1 (shift), \mathcal{N}_2 (swap) e \mathcal{N}_3 (2-swap).
- 3) Enquanto não for atingido o critério de parada:
 - a) Aplicar um *shaking* em \mathcal{N}_k para obter x' .
 - b) Aplicar busca local Best Improvement na vizinhança shift, obtendo x'' .
 - c) Se x'' melhora o valor da função escalar:
 - atualizar $x \leftarrow x''$ e reiniciar em \mathcal{N}_1 ;
 - d) caso contrário:
 - avançar para a próxima vizinhança \mathcal{N}_{k+1} (ou voltar a \mathcal{N}_1 se $k = 3$).

Como critério de parada, foi utilizado um número máximo de iterações sem melhoria e um limite de iterações totais.

VI. RESULTADOS DA OTIMIZAÇÃO MONO-OBJETIVO

A. Resultados numéricos

A Tabela I apresenta os resultados para a minimização de f_1 (custo total), em cinco execuções independentes do VNS.

Execução	1	2	3	4	5
Valor	895	912	894	909	898
Mínimo:	894			Máximo: 912	
Média:	901,6			Desvio padrão: 7,45	

TABLE I

RESULTADOS ESTATÍSTICOS PARA f_1 (CUSTO TOTAL).

Para a minimização de f_2 (desequilíbrio de carga), obtiveram-se os valores mostrados na Tabela II.

Execução	1	2	3	4	5
Valor	1	2	1	0	2
Mínimo:	0			Máximo: 2	
Média:	1,2			Desvio padrão: 0,75	

TABLE II

RESULTADOS ESTATÍSTICOS PARA f_2 (DESEQUILÍBRIOS DE CARGA).

B. Curvas de convergência

As Figuras 1 e 2 apresentam as curvas de convergência (cinco execuções sobrepostas) para a minimização de f_1 e f_2 , respectivamente.

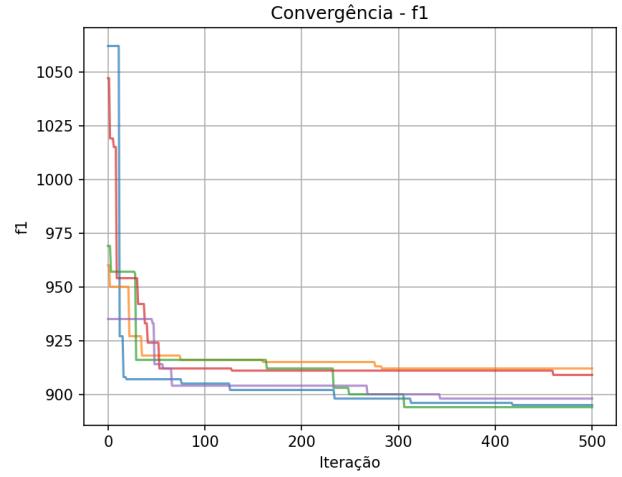


Fig. 1. Curvas de convergência para f_1 (custo total) em 5 execuções do VNS.

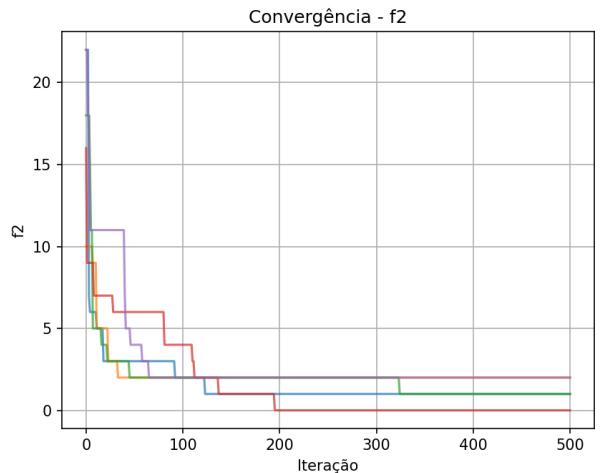


Fig. 2. Curvas de convergência para f_2 (desequilíbrio de carga) em 5 execuções do VNS.

C. Melhor solução encontrada

As Figuras 3 e 4 apresentam a distribuição de carga por agente na melhor solução encontrada para cada função objetivo.

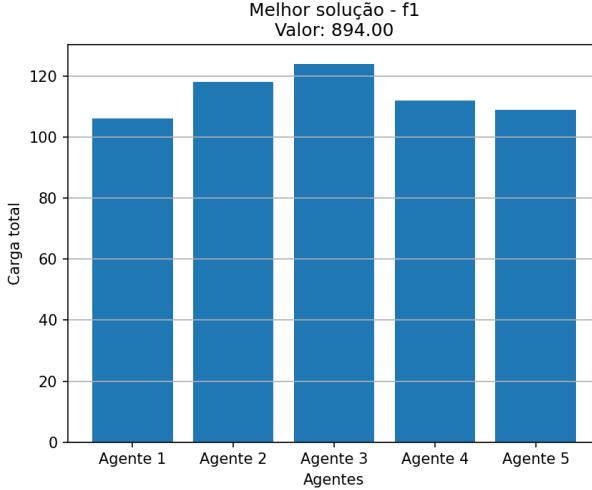


Fig. 3. Melhor solução encontrada para f_1 (custo mínimo).

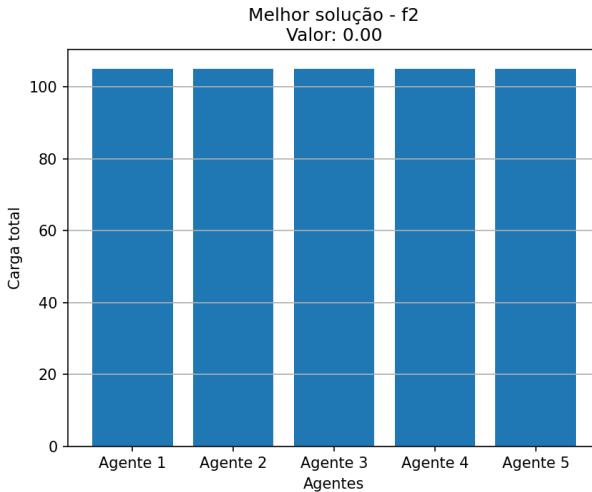


Fig. 4. Melhor solução encontrada para f_2 (carga mais equilibrada).

Os resultados confirmam o caráter conflitante dos objetivos. A solução de menor custo observado (894) apresenta um desequilíbrio considerável (maior que 20 unidades, dependendo da execução), enquanto a solução com desequilíbrio perfeito (0) exige um custo significativamente mais elevado (acima de 1100). Essa disparidade justifica a necessidade de uma abordagem multiobjetivo para encontrar soluções de compromisso.

VII. OTIMIZAÇÃO MULTIOBJETIVO

Os resultados da otimização mono-objetivo confirmaram a natureza conflitante das funções $f_1(x)$ (custo) e $f_2(x)$ (equilíbrio). A solução de menor custo obtida nas execuções mono-objetivo apresenta um desequilíbrio significativo, enquanto soluções com desequilíbrio praticamente nulo não são, em geral, as de menor custo.

O objetivo desta segunda etapa é encontrar o conjunto de soluções de compromisso entre esses dois extremos, conhecido como **fronteira de Pareto**. Para gerar esta fronteira, o problema multiobjetivo é transformado em um problema mono-objetivo por meio de métodos de escalarização [4]. Isso permite que o mesmo algoritmo VNS desenvolvido na etapa anterior seja utilizado para encontrar soluções não-dominadas.

Conforme solicitado, foram implementadas duas abordagens escalares: a Soma Ponderada (P_w) e o Método ϵ -Restrito (PE).

A. Abordagem escalar: Soma Ponderada (P_w)

A abordagem de Soma Ponderada combina as funções objetivo em uma única função escalar $f_{Pw}(x)$, por meio de uma soma ponderada:

$$f_{Pw}(x) = w_1 \cdot \hat{f}_1(x) + w_2 \cdot \hat{f}_2(x), \quad (7)$$

em que $w_1, w_2 \geq 0$ são os pesos atribuídos a cada objetivo, com $w_1 + w_2 = 1$, e \hat{f}_1, \hat{f}_2 são versões normalizadas das funções objetivo.

Normalização: As funções $f_1(x)$ e $f_2(x)$ operam em escalas diferentes: enquanto $f_1(x)$ está na ordem de centenas (por exemplo, entre 900 e 1100), $f_2(x)$ varia tipicamente em poucas unidades (por exemplo, entre 0 e 10). Se aplicássemos a soma ponderada diretamente, o termo de custo dominaria a soma, mesmo com peso pequeno.

Para mitigar esse efeito, as funções objetivo são **normalizadas** para o intervalo $[0, 1]$. A normalização $\hat{f}_k(x)$ de uma função $f_k(x)$ ($k = 1, 2$) é dada por:

$$\hat{f}_k(x) = \frac{f_k(x) - f_k^{\min}}{f_k^{\max} - f_k^{\min}}, \quad (8)$$

em que f_k^{\min} e f_k^{\max} são estimados a partir das melhores soluções obtidas nas otimizações mono-objetivo de f_1 e f_2 (valores ideal e nadir aproximados).

A função escalar final a ser minimizada pelo VNS torna-se, portanto:

$$\text{Minimizar } f_{Pw}(x) = w_1 \cdot \hat{f}_1(x) + w_2 \cdot \hat{f}_2(x). \quad (9)$$

Para gerar a fronteira, o VNS foi executado múltiplas vezes, variando-se os pesos w_1 e w_2 (com $w_1 + w_2 = 1$) em passos discretos, de forma a obter aproximadamente 20 soluções não-dominadas ao longo da fronteira.

B. Abordagem escalar: Método ϵ -Restrito (PE)

O Método ϵ -Restrito transforma o problema multiobjetivo em um mono-objetivo de forma diferente: escolhe-se uma função objetivo prioritária para otimização e trata-se a outra como uma nova restrição ao problema.

Neste trabalho, escolheu-se minimizar a função de custo $f_1(x)$, tratando $f_2(x)$ como restrição. O novo modelo matemático é:

$$\text{Minimizar } f_1(x)$$

Sujeito a:

$$\sum_{i=1}^m x_{ij} = 1, \quad \forall j \in \mathcal{T} \quad (\text{atribuição única}) \quad (10)$$

$$\sum_{j=1}^n a_{ij} \cdot x_{ij} \leq b_i, \quad \forall i \in \mathcal{A} \quad (\text{capacidade}) \quad (11)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in \mathcal{A}, \forall j \in \mathcal{T} \quad (\text{domínio}) \quad (12)$$

$$f_2(x) \leq \epsilon_2, \quad (\text{restrição } \epsilon \text{ de desequilíbrio}). \quad (13)$$

O parâmetro ϵ_2 representa o valor máximo aceitável para o desequilíbrio de carga. A fronteira de Pareto é gerada executando o VNS múltiplas vezes, variando-se o valor de ϵ_2 . Neste trabalho, ϵ_2 foi discretizado em aproximadamente 20 níveis igualmente espaçados, variando entre o desequilíbrio mínimo (próximo de 0) e o desequilíbrio observado na solução de custo mínimo.

A principal vantagem desta abordagem é que ela não sofre do problema de escalas e, portanto, **dispensa a normalização explícita** das funções objetivo.

C. Resultados da Otimização Multiobjetivo

Para cada uma das duas abordagens (Pw e PE), o processo de otimização foi executado 5 vezes. Em cada execução, cerca de 20 pontos da fronteira foram estimados, e os resultados das 5 execuções foram sobrepostos. Em seguida, um conjunto único de soluções não-dominadas foi filtrado de todos os pontos encontrados, e 20 soluções representativas foram selecionadas para compor a “Fronteira Final”.

1) Resultados da Soma Ponderada (Pw): A Figura 5 apresenta os resultados obtidos pela abordagem da Soma Ponderada. As 5 execuções distintas estão representadas por cores diferentes, e a fronteira final agregada é marcada com um “X” em destaque.

É possível observar que o método foi bem-sucedido em encontrar soluções ao longo da curva de Pareto, demonstrando o conflito entre o custo (f_1) e o desequilíbrio (f_2). No entanto, a distribuição dos pontos é visivelmente irregular: há um agrupamento de soluções nos extremos da fronteira (regiões de baixo custo/alto desequilíbrio e alto custo/baixo desequilíbrio), com menos soluções na região de “joelho” da curva. Esse comportamento é consistente com a teoria, que aponta dificuldades da Soma Ponderada em capturar partes não convexas da fronteira.

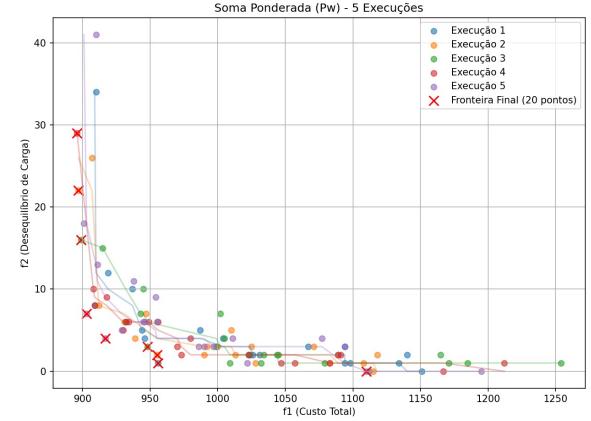


Fig. 5. Resultados das 5 execuções da abordagem Soma Ponderada (Pw).

2) Resultados do Método ϵ -Restrito (PE): A Figura 6 apresenta os resultados da abordagem do Método ϵ -Restrito. Novamente, as 5 execuções estão sobrepostas, compondo a fronteira final.

Em contraste com a Soma Ponderada, o Método ϵ -Restrito gerou uma fronteira de Pareto com distribuição de pontos significativamente mais uniforme e completa. Ao variar o valor da restrição ϵ_2 em passos discretos, o VNS é “forçado” a encontrar a solução de menor custo para cada nível de desequilíbrio permitido, o que leva a um mapeamento bem mais granular de toda a curva de *trade-off*, incluindo a região de joelho.

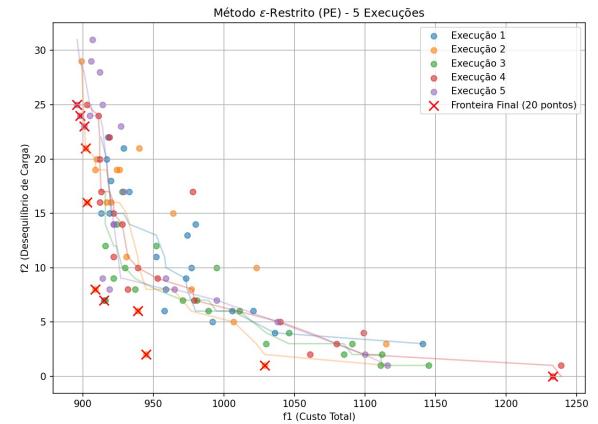


Fig. 6. Resultados das 5 execuções da abordagem Método ϵ -Restrito (PE).

3) Comparação das abordagens: Ambos os métodos foram capazes de estimar a fronteira de Pareto, validando a eficácia do VNS como algoritmo de otimização. No entanto, a análise comparativa dos gráficos demonstra uma clara vantagem qualitativa do Método ϵ -Restrito (PE) para esta instância: o PE gerou um conjunto de soluções não-dominadas mais rico e melhor distribuído do que a Soma Ponderada.

Devido à melhor qualidade e cobertura da fronteira, a "Fronteira Final" utilizada na etapa de tomada de decisão multicritério foi construída a partir das soluções não-dominadas obtidas com o Método ϵ -Restrito.

VIII. TOMADA DE DECISÃO MULTICRITÉRIO

A terceira etapa empregou a fronteira final produzida pelo Método ϵ -Restrito (PE) como conjunto de alternativas candidatas. A partir dela, foram selecionadas 10 soluções não-dominadas representativas.

A. Critérios e pesos

Para cada solução foram avaliados quatro atributos, combinando desempenho nominal e robustez:

- f_1 : custo total (critério de custo);
- f_2 : desequilíbrio de carga (critério de custo);
- **Folga mínima**: $\min_i \{b_i - \text{carga}_i\}$ (critério de benefício);
- **Variação de custo**: média da variação absoluta do custo sob perturbações aleatórias de $\pm 10\%$ na matriz c (critério de custo). As mesmas simulações registraram ainda folga média e violação esperada de capacidade, usadas como diagnóstico de robustez.

Os pesos consensualados foram

$$w = [0,35, 0,25, 0,20, 0,20]$$

para f_1 , f_2 , folga mínima e variação de custo, respectivamente. Os tipos (custo/benefício) determinaram o sentido da normalização.

Critério	Tipo	Peso
f_1 (custo total)	Custo	0,35
f_2 (desequilíbrio)	Custo	0,25
Folga mínima	Benefício	0,20
Variação de custo	Custo	0,20

TABLE III

CRITÉRIOS E PESOS USADOS NA DECISÃO MULTICRITÉRIO.

A escolha desses quatro critérios buscou equilibrar o desempenho nominal (custo e equilíbrio) com medidas de robustez, refletindo a preferência por soluções que permaneçam aceitáveis sob incertezas nos custos.

B. Procedimentos de avaliação

a) *Aggregação clássica ponderada*.: Após normalização min–max de cada coluna, segundo o tipo de critério, a nota final foi computada por soma ponderada:

$$S_{\text{clássico}}(x) = \sum_{k=1}^4 w_k \tilde{f}_k(x), \quad (14)$$

em que $\tilde{f}_k(x)$ representa a forma normalizada do critério k (para critérios de custo, valores menores recebem melhores notas na escala normalizada).

b) *TOPSIS*.: Também foi aplicado o método **TOPSIS** [5]. Os dados foram normalizados pela norma euclidiana coluna a coluna, ponderados por w_k e comparados aos pontos ideal (f_k^+) e nadir (f_k^-) de cada critério. A pontuação corresponde ao coeficiente de proximidade à solução ideal:

$$C^*(x) = \frac{D^-(x)}{D^+(x) + D^-(x)}, \quad (15)$$

em que D^+ e D^- são as distâncias euclidianas aos vetores ideal e nadir, considerando o sentido custo/benefício de cada componente.

C. Síntese dos resultados

Ambos os métodos geraram rankings coerentes, concentrados na região intermediária da fronteira (região de joelho). As soluções selecionadas como mais atrativas foram:

- **Escolha clássica** (maior $S_{\text{clássico}}$): $f_1 = 946,0$, $f_2 = 2,0$, folga mínima 1,4, variação de custo 5,68.
- **Escolha TOPSIS** (maior C^*): $f_1 = 1042,0$, $f_2 = 1,0$, folga mínima 2,4, variação de custo 9,13.

A solução clássica privilegia um custo menor, aceitando um desequilíbrio ligeiramente maior e uma folga mínima mais apertada. Já a solução TOPSIS privilegia um equilíbrio de carga quase perfeito (desequilíbrio igual a 1) e folga maior, ao custo de um aumento relevante em f_1 e de maior sensibilidade às perturbações de custo.

Ambas as soluções mantiveram violações esperadas de capacidade praticamente nulas nos cenários simulados, indicando que os trade-offs se concentram, principalmente, entre custo, equilíbrio e robustez de custo.

Solução	f_1	f_2	Folga mín.	Var. custo
Escolha clássica (Pw, $w_1 = 0,84$)	946,0	2,0	1,4	5,68
Escolha TOPSIS (Pw, $w_1 = 0,32$)	1042,0	1,0	2,4	9,13

TABLE IV
SOLUÇÕES DESTACADAS PELOS MÉTODOS DE DECISÃO.

Em termos práticos, a solução clássica é mais indicada para decisões com forte aversão a custo, que ainda toleram leve desequilíbrio. A solução TOPSIS é mais atraente para decisões sensíveis a desequilíbrios de carga, dispostos a pagar um custo maior e conviver com maior variabilidade de custo.

D. Análise Visual da Decisão

Para complementar a análise numérica, as visualizações gráficas consolidam o processo decisório. A Fig. 7 posiciona as soluções escolhidas (Clássica e TOPSIS) dentro da fronteira de Pareto, evidenciando visualmente o *trade-off* entre custo e equilíbrio: a solução TOPSIS situa-se na região de "joelho" (maior equilíbrio), enquanto a Clássica desloca-se para a esquerda (menor custo).

A Fig. 8 compara as pontuações obtidas pelas principais alternativas nos dois métodos de ranking, corroborando a robustez das escolhas. Por fim, a Fig. 9 detalha a alocação de recursos por agente, permitindo verificar a homogeneidade

da distribuição de carga na solução TOPSIS em contraste com a leve assimetria da solução Clássica.

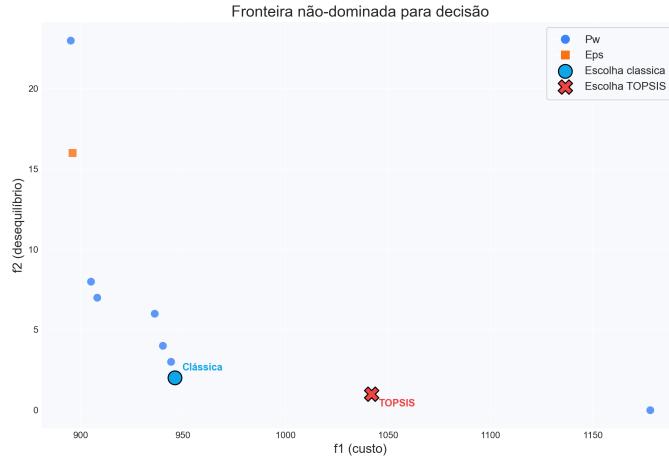


Fig. 7. Fronteira não-dominada usada na decisão multicritério, com destaque para as soluções escolhidas (clássica e TOPSIS).



Fig. 8. Escores das alternativas segundo a agregação clássica ponderada e o TOPSIS.

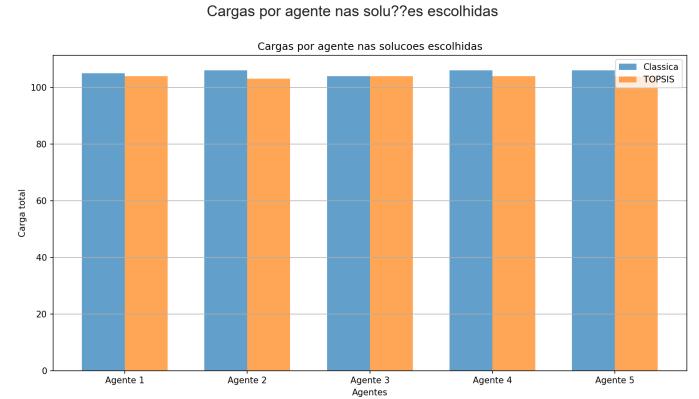


Fig. 9. Distribuição de carga por agente para as soluções selecionadas.

IX. CONCLUSÕES

Este trabalho abordou o problema de atribuição de tarefas a agentes sob uma ótica multiobjetivo, focando no conflito entre a minimização do custo total (f_1) e a minimização do desequilíbrio de carga (f_2).

Na primeira etapa, foi desenvolvido um algoritmo baseado na metaheurística **Variable Neighborhood Search (VNS)** combinando uma heurística construtiva do tipo **GRASP** com busca local **Best Improvement** na vizinhança *shift*. Os resultados mono-objetivo demonstraram a eficiência do algoritmo ao produzir soluções de baixo custo e alto equilíbrio, servindo como referência para as etapas posteriores.

Na segunda etapa, o VNS foi aplicado a duas abordagens de escalarização para gerar a fronteira de Pareto. A **Soma Ponderada (Pw)** foi capaz de encontrar soluções não-dominadas, mas gerou uma fronteira relativamente irregular, concentrando pontos nos extremos. O **Método ϵ -Restrito (PE)**, por sua vez, produziu uma fronteira significativamente mais uniforme e bem distribuída, incluindo a região de joelho, e foi, por isso, adotado como base para a etapa de decisão.

Na etapa decisória, foram considerados simultaneamente custo, equilíbrio e medidas de robustez a incertezas nos custos. A combinação de métodos de decisão multicritério (agregação clássica ponderada e TOPSIS) levou a duas recomendações complementares: (i) uma solução de menor custo, com leve desequilíbrio e folga mais apertada, e (ii) uma solução com equilíbrio quase perfeito e folga maior, porém custo mais elevado e maior sensibilidade a perturbações de custo. Ambas mantiveram violações de capacidade desprezíveis nas simulações, oferecendo perfis claros para decisores com diferentes aversões a custo ou a desequilíbrio.

Conclui-se que o VNS, combinado com escalarizações adequadas e um processo de decisão multicritério bem estruturado, constitui uma ferramenta poderosa para apoiar a escolha de soluções de compromisso em problemas de atribuição com restrições de capacidade.

REFERENCES

- [1] P. Hansen and N. Mladenović, “Variable neighborhood search: Principles and applications,” *European Journal of Operational Research*, vol. 130, no. 3, pp. 449–467, 2001.

- [2] G. R. Mateus, M. G. C. Resende, and R. M. A. Silva, "Grasp with path-relinking for the generalized quadratic assignment problem," *Journal of Heuristics*, vol. 17, pp. 527–565, 2011.
- [3] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, 2001.
- [4] R. Takahashi, "Otimização escalar e vetorial," Notas de aula, Conceitos Preliminares, Engenharia de Controle e Automação, UFMG, 2025.
- [5] C. L. Hwang and K. Yoon, *Multiple Attribute Decision Making: Methods and Applications*. New York: Springer-Verlag, 1981.