

Método Multicritério Classificatório THOR-Sort: Tutorial do Software

Fernanda Márcia da Silveira Esteves
esteves.fernandamarcia@gmail.com

UFF

Carlos Francisco Simões Gomes
carlos_gomes@id.uff.br

UFF

Diogo Ferreira de Lima Silva
diogofls@id.uff.br

UFF

Marcos dos Santos
marcosdossantos_doutorado_uff@yahoo.com.br

IME

Luiz Flávio Autran Monteiro Gomes
lfautranmg@gmail.com

UFABC

Resumo: A tomada de decisão em cenários complexos exige a análise de múltiplos critérios, por vezes conflitantes entre si, diante de diversas alternativas. Nesse cenário, destaca-se a utilização de Métodos de Apoio Multicritério à Decisão. O objetivo deste artigo é apresentar o THOR-Sort, um novo método multicritério que classifica alternativas em processos decisórios complexos, bem como demonstrar sua validação e um tutorial de seu software. Os resultados demonstraram a efetividade do método, revelando assertividade, confiabilidade e consistência nas classificações obtidas. Destaca-se a importância do THOR-Sort em fornecer uma classificação detalhada e adequada das alternativas, especialmente em situações onde a ordenação simples é insuficiente. Além disso, o artigo proporciona um tutorial detalhado do software, facilitando a compreensão e utilização do THOR-Sort pela comunidade acadêmica e por organizações públicas e privadas. Este tutorial, ao exemplificar a aplicação do método, busca ampliar o conhecimento e incentivar a adoção do THOR-Sort em diversos contextos práticos. Este estudo, ao demonstrar que os resultados gerados pelo THOR-Sort são consistentes com métodos anteriores, reforça a aderência e eficácia do novo método. Em suma, o THOR-Sort se apresenta como uma ferramenta robusta e confiável para a classificação de alternativas, trazendo avanços significativos para a área de tomada de decisão multicritério.

Palavras Chave: Tomada de Decisão - Método Multicritério - THOR-Sort - Classificação - Tutorial de Software

1. INTRODUÇÃO

A vida de todos os indivíduos é repleta de situações em que é preciso tomar decisões, tanto no âmbito pessoal quanto profissional. Segundo Gomes *et al.* (2020), os problemas de decisão do mundo real raramente contêm um único critério, apresentando alta complexidade ao buscar uma solução adequada. Nesse contexto, os Métodos de Apoio Multicritério à Decisão (AMD) são implementados a fim de que sejam feitas as melhores escolhas possíveis, considerando todo o contexto da situação, as alternativas e os critérios envolvidos.

De acordo com Infante *et al.* (2017), os métodos AMD têm sido desenvolvidos para apoiar e conduzir decisores na avaliação e escolha de alternativas consideradas soluções para determinado problema. Ao longo dos anos, com o aumento de sua utilização, tem-se desenvolvido e aperfeiçoado os métodos existentes.

Conforme Esteves *et al.* (2022), um dos grandes desafios da área de Suprimentos de empresas de grande porte é a priorização de demandas, frente ao alto volume de solicitações recebidas diariamente e à quantidade reduzida de mão de obra. Portanto, é necessária uma estratégia de atendimento, a fim de que todas as demandas sejam atendidas em tempo hábil e com a qualidade requerida. Para tal, são necessárias técnicas e ferramentas que permitam análises e auxiliem as tomadas de decisão.

Mais eficaz do que ordenar as demandas, seria classificá-las, pois classificando as requisições em “Alta Prioridade”, “Média Prioridade” e “Baixa Prioridade” é possível distribuir, entre os funcionários, demandas de diversos níveis, sem que nenhum deles fique sobrecarregado por ter mais demandas urgentes do que os demais. E, ainda, possibilita que os compradores saibam quais requisições devem priorizar dentro de sua carteira. Sem essa classificação, não é possível saber com clareza qual é a prioridade de cada demanda, principalmente das que não estão entre as primeiras e as últimas de uma lista ordenada. Além disso, a classificação possibilita a tomada de decisão a nível gerencial a longo prazo.

Neste trabalho é apresentado o THOR-Sort, um novo método AMD para a classificação de alternativas em processos decisórios, que integra critérios quantitativos e qualitativos de forma robusta, podendo lidar com incertezas. Foi desenvolvida uma ferramenta computacional para esse método, na linguagem de programação Python. Neste artigo é demonstrado um tutorial desse *software*, que foi utilizado na validação do método THOR-Sort.

Este artigo possui seis capítulos, sendo o primeiro de caráter introdutório. No segundo capítulo aborda-se o referencial teórico, contendo uma visão geral sobre os métodos AMD e os métodos THOR e THOR 2. No terceiro capítulo é exposto o algoritmo do THOR-Sort. No quarto capítulo é demonstrada a validação do método. No quinto capítulo consta o tutorial do *software*. E no sexto capítulo são apresentadas as conclusões sobre o tema.

2. REFERENCIAL TEÓRICO

2.1. APOIO MULTICRITÉRIO À DECISÃO

Em ambientes empresariais cada vez mais complexos, com problemas organizacionais desafiadores, é imprescindível contar com métodos estruturados que possam oferecer suporte nas tomadas de decisão. A Pesquisa Operacional (PO) é a área de conhecimento que se dedica ao estudo, desenvolvimento e aplicação de métodos analíticos avançados, com o propósito de contribuir para decisões mais assertivas em variadas áreas de atuação humana. Com técnicas de modelagem matemática e eficientes algoritmos computacionais, a PO pode auxiliar o decisor na análise dos mais variados aspectos e situações de um problema complexo, permitindo a tomada de decisão efetiva e a construção de sistemas mais produtivos (SOBRAPO, 2022).

Na PO, o AMD é o ramo que permite a estruturação e compreensão de problemas em ambientes complexos, que exigem a avaliação de múltiplas variáveis. Desta forma, os métodos AMD são técnicas que auxiliam o agente decisor na tomada de decisão sobre problemas não triviais, avaliando, ordenando, classificando e selecionando alternativa(s) dentro de critérios pré-definidos (SILVA *et al.*, 2018).

Conforme Gomes *et al.* (2003), para obter sucesso nos negócios as empresas necessitam de profissionais qualificados que, após considerarem as diferentes alternativas, sejam capazes de tomar decisões utilizando os vários critérios existentes. Para facilitar esse processo, nas décadas de 60 e 70, foram desenvolvidos os primeiros métodos AMD, destinados a resolver problemas em que múltiplos objetivos precisavam ser alcançados simultaneamente. Esses métodos possuem natureza científica e subjetiva, permitindo a consideração de características não quantitativas que auxiliam no processo de tomada de decisão.

Com relação aos problemas de decisão, foram propostas por Roy (1968) as problemáticas a seguir, que não são independentes entre si:

- Tipo α - Problema de seleção: Envolve a escolha de uma única alternativa entre várias opções disponíveis, com base em critérios predefinidos.
- Tipo β - Problema de classificação: Consiste na categorização das alternativas de acordo com critérios específicos, sem, necessariamente, estabelecer uma ordem.
- Tipo γ - Problema de ordenação: Compreende a disposição ordenada das alternativas de acordo com a sua preferência relativa em relação aos critérios estabelecidos.
- Tipo δ - Problema de descrição: Visa esclarecer a decisão ao descrever as características das alternativas e seus impactos nos critérios.

Os métodos AMD podem ser classificados de diversas maneiras, porém, a de maior relevância é a classificação entre Escola Americana e Escola Francesa. A primeira tem um conjunto de métodos voltados à utilização da Teoria da Utilidade Multiatributo, que é uma abordagem para a tomada de decisão com múltiplos critérios ou atributos relevantes para uma escolha, envolvendo a análise e comparação sistemática de alternativas, e um outro conjunto, denominado Métodos Iterativos, que permite a participação ativa dos tomadores de decisão ao longo do processo, permitindo ajustes e refinamentos com base em feedback contínuo durante a análise. A Escola Francesa, por sua vez, não tem como objetivo a obtenção de um critério único de síntese. Por isso, desenvolve relações de sobreclassificação para auxiliar os tomadores de decisão no confronto entre as alternativas em uma abordagem por pares, permitindo que as preferências sejam ranqueadas de formas distintas (TENÓRIO *et al.*, 2019).

Segundo Tenório (2020), os métodos AHP - *Analytic Hierarchy Process* (1980), ANP - *Analytic Network Process* (2006), MAUT - *Multiple Attribute Utility Theory* (1993), MACBETH - *Measuring Attractiveness by a Categorical Based Evaluation Technique* (1994), SMART - *Simple Multi Criteria Attribute Rating Technique* (1977), TODIM - *Tomada de Decisão Interativa Multicritério* (1991) e TOPSIS - *Technique for Order of Preference by Similarity to Ideal Solution* (1981) seguem a linha da Escola Americana. Já os métodos da Família ELECTRE - *Elimination and Choice Translating Reality for Enrichment Evaluation* (1968) e PROMETHEE - *Preference Ranking Method for Enrichment Evaluation* (1985) seguem os princípios da Escola Europeia.

Embora Rangel e Gomes (2010) sugerirem que os métodos MACBETH e TODIM possuem elementos técnicos de ambas as escolas, os próprios autores do MACBETH afirmam que o método é associado à escola americana. Já os métodos THOR (1999) e o THOR 2 (2020) são classificados pelos seus próprios autores como híbridos (GOMES *et al.*, 2008).

2.1. OS MÉTODOS THOR E THOR 2

O método AMD THOR foi proposto por Gomes (1999) e combina elementos da modelagem de preferência (assemelhando-se à abordagem da Escola Francesa) e das Teorias Multiatributo e de Utilidade (alinhando-se com a abordagem da Escola Americana). Segundo Gomes *et al.* (2008), esse método apresenta uma série de vantagens significativas, tais como: quantificação de incertezas, eliminação de critérios irrelevantes, aplicável a processos de decisão transitivos e não transitivos, facilidade de operação e rápida geração de soluções, permite a entrada simultânea de dados de múltiplos decisores.

Essas vantagens tornam o THOR uma ferramenta valiosa para análise e tomada de decisões, especialmente em situações complexas. Entretanto, para aplicar a metodologia THOR, é necessário estabelecer os seguintes pontos: atribuir peso para cada critério, representando a importância relativa deles em relação aos outros; definir limite de preferência (p) e limite de indiferença (q) para cada critério; estabelecer valor de discordância para cada critério; determinar pertinência dos pesos e da classificação das alternativas.

Na aplicação do método THOR, dadas duas alternativas a e b , três cenários são analisados: S1, S2 e S3. Comparando as alternativas, verificam-se os critérios onde ocorrem aPb (preferência forte de a sobre b), aQb (preferência fraca de a em relação a b), aIb (indiferença entre as alternativas) e discordância. De acordo com Rangel e Gomes (2010), nas situações S2 e S3, ocorre um cenário mais flexível, em que uma pequena diferença entre as alternativas em cada critério é suficiente para classificar uma alternativa como superior à outra.

As associações P, I e Q são expostas nas Equações 2.1.1, 2.1.2 e 2.1.3, respectivamente.

$$aPb \leftrightarrow g(a) - g(b) > p \quad (2.1.1)$$

$$aIb \leftrightarrow -q \leq g(a) - g(b) \leq q \quad (2.1.2)$$

$$aQb \leftrightarrow q < |g(a) - g(b)| \leq p \quad (2.1.3)$$

Onde,

$g(a)$: Valor da alternativa a dentro de um critério;

$g(b)$: Valor da alternativa b dentro de um critério.

E as situações S1, S2 e S3 podem ser visualizadas nas Equações 2.1.4, 2.1.5 e 2.1.6, respectivamente.

$$S1 : \sum_{j=1}^n (w_j | aP_jb) > \sum_{j=1}^n (w_j | aQ_jb + aI_jb + aR_jb + bQ_ja + bP_ja) \quad (2.1.4)$$

$$S2 : \sum_{j=1}^n (w_j | aP_jb + aQ_jb) > \sum_{j=1}^n (w_j | aI_jb + aR_jb + bQ_ja + bP_ja) \quad (2.1.5)$$

$$S3 : \sum_{j=1}^n (w_j | aP_jb + aQ_jb + aI_jb) > \sum_{j=1}^n (w_j | aR_jb + bQ_ja + bP_ja) \quad (2.1.6)$$

O método THOR passou por uma evolução axiomática resultando no método THOR 2 (TENÓRIO, 2020). Essa evolução envolveu modificações no algoritmo original, com base em uma análise conjunta com a revisão dos principais métodos das escolas americana e francesa. As principais diferenças do método THOR 2 em relação ao original são as seguintes:

- Distinção na atribuição dos pesos no somatório de pontuações para $aIjb$ e $aQjb$ em S1, S2 e S3:

- Para as comparações em que ocorre $aIjb$, é atribuída metade do peso do respectivo critério;
- Para as comparações em que ocorre $aQjb$, é utilizada uma proporção entre metade do peso do critério ($aIjb$) e o peso total ($aPjb$), conforme a Equação 2.1.7:

$$peso_i * (((a_i - q_i) / (p_i - q_i)) * 0.5 + 0.5) \quad (2.1.7)$$

- O THOR original considera apenas a multiplicação pelo índice na situação $aQjb$, degradando apenas nesse caso. Já o THOR 2 inclui a depreciação da pontuação nas situações de preferência forte ($aPjb$) e indiferença ($aIjb$).
- No THOR 2, toda a incerteza presente na atribuição das classificações das alternativas e dos pesos é quantificada.

Essas modificações no THOR 2 permitem uma abordagem mais refinada para lidar com a atribuição dos pesos, a depreciação da pontuação e a quantificação da incerteza, melhorando a precisão e a robustez do método em relação ao THOR original.

3. THOR-SORT

O THOR-Sort representa uma variação importante do método THOR 2, uma vez que esse último é utilizado em problemas tipo γ , promovendo uma ordenação das alternativas. Já essa nova ferramenta, é utilizada em problemas do tipo β , classificando ordinalmente as alternativas. As definições e o algoritmo do THOR-Sort podem ser visualizados a seguir (ESTEVEES, 2024).

Seja $G = \{g_1, \dots, g_n\}$ um conjunto de n critérios, sendo que para todos os critérios de custo, as avaliações de alternativas e perfis de contorno devem ser multiplicadas por (-1).

Seja $p = [p_1, \dots, p_n]$, $q = [q_1, \dots, q_n]$ e $D = [D_1, \dots, D_n]$ respectivamente, limites de preferência, indiferença e discordância sobre os critérios.

Seja $w = [w_1, w_2, \dots, w_n]$ um vetor de pesos que representam a importância relativa dos critérios.

Seja $C = \{C_1, \dots, C_l\}$ um conjunto de classes l pré-estabelecidas e ordenadas, onde $C_1 > C_2 > \dots > C_l$.

Seja $A = \{a_1, \dots, a_m\}$ um conjunto de m alternativas. Esse conjunto pode ser representado por uma Matriz de Decisão $A = [a_{i,j}]_{m \times n}$, onde $a_{i,j}$ representa a avaliação da alternativa a_i em relação ao critério g_j .

Seja $B = \{b_1, \dots, b_{l-1}\}$ um conjunto de perfis de fronteira. Um perfil de fronteira é construído no limite entre cada duas classes consecutivas. Por exemplo, b_1 é um perfil limitante para as classes C_1 e C_2 . O conjunto de perfis pode ser representado por uma matriz $B = [b_{k,j}]_{(l-1) \times n}$, onde $b_{k,j}$ representa a avaliação do perfil b_k em relação ao critério g_j .

Sejam $\Gamma_A = [\alpha_{i,j}]_{m \times n}$, $\Gamma_B = [\beta_{k,j}]_{(l-1) \times n}$ e $\omega = [\omega_1, \omega_2, \dots, \omega_n]$, respectivamente, os valores de pertinência referentes à A , B , e w . Se o decisor não optar por trabalhar com valores de pertinência, basta inserir os valores iguais a 1 nessas matrizes.

3.1. ALGORITMO THOR-SORT

Os passos do método THOR-Sort foram implementados na linguagem de programação Python. O código computacional está disponível por meio do link: <https://github.com/fernandaesteves/THORSORT>.

O algoritmo do THOR-Sort é descrito a seguir.

1º Passo: Testar a consistência dos perfis de contorno.

1.1 Para que seja consistente, $b_{k,j} \geq b_{k+1,j}, \forall k, j$.

2º Passo: Para cada alternativa a_i :

2.1 Concatena-se verticalmente a linha da matriz A correspondente à alternativa $A_{i,j}$ e B , resultando na seguinte matriz:

$$\begin{bmatrix} a_{i,1} & a_{i,2} & \dots & a_{i,n} \\ b_{1,1} & b_{1,2} & \dots & b_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{l-1,1} & b_{l-1,2} & \dots & b_{l-1,n} \end{bmatrix}$$

2.2 Concatena-se da mesma forma os valores de pertinência para a alternativa e os perfis $\Gamma_{A_{i,j}}$ e Γ_B , resultando na seguinte matriz:

$$\begin{bmatrix} \alpha_{i,1} & \alpha_{i,2} & \dots & \alpha_{i,n} \\ \beta_{1,1} & \beta_{1,2} & \dots & \beta_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{l-1,1} & \beta_{l-1,2} & \dots & \beta_{l,n} \end{bmatrix}$$

2.3 Para cada critério, deve-se construir a seguinte matriz relacional, por exemplo:

<i>Criterion j</i>	a_i	b_1	b_2	...	b_l
a_i	—	$b_1 P a_i$	$a_i I b_2$	...	$a_i P b_l$
b_1	—	—	$b_1 Q b_2$	...	$b_1 P b_l$
b_2	—	—	—	...	$b_2 P b_l$
$\vdots$	—	—	—	$\ddots$	$\vdots$
b_l	—	—	—	—	—

2.4 Para cada critério, seguindo as regras do THOR 2 para as comparações pareadas, deve-se construir a tabela relacional de valores. O valor de $v_{u,v}$ depende da relação entre alternativas/perfis u e v obtido na Etapa 2.3. Para o critério j , tem-se a matriz $R_j = [r_{j_{u,v}}]_{l \times l}$.

<i>Criterion j</i>	a_i	b_1		b_2	...	b_l
a_i	0	0		$\frac{w_j}{2}$	...	w_j
b_1	w_j	0	$w_j \times \left(\frac{(b_{1,j} - b_{2,j}) - q_j}{p_j - q_j} \times 0.5 + 0.5 \right)$		...	w_j
b_2	$\frac{w_j}{2}$	0		0	...	w_j
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\ddots$	$\vdots$
b_l	0	0		0	...	0

2.5 Multiplicar cada valor das matrizes obtidas na Etapa 2.4 pelo valor de pertinência correspondente. Para alternativa i , perfil k e critério j , o valor é obtido pela seguinte equação:

$$\frac{\alpha_{i,j} + \beta_{k,j} + \omega_j}{3}$$

Nessa etapa, um total de n matrizes do tipo $V_j = [v_{j,u,v}]_{l \times l}$ será calculado.

2.6 Verificar, nas comparações pareadas, os cenários S_1 , S_2 e S_3 .

Cenário S_1 :

$$a_i T_{S_1} b_k$$

$$SE \left[\sum_{j=1}^n (t_j | a_i P_j b_k) > \sum_{j=1}^n (t_j | a_i Q_j b_k \cup a_i I_j b_k \cup a_i R_j b_k \cup b_k Q_j a_i \cup b_k P_j a_i) \right]$$

$$E [(b_{k,j} - a_{i,j}) < d_j, \forall j]$$

Se a alternativa/perfil da linha i dominar a alternativa/perfil da coluna j , deve-se inserir “T” na célula correspondente da matriz a seguir.

$$T_1: \begin{array}{c|cccccc} & a_i & b_1 & b_2 & \dots & b_l \\ \hline a_i & \square & \square & \square & \dots & \square \\ b_1 & \square & \square & \square & \dots & \square \\ b_2 & \square & \square & \square & \dots & \square \\ \vdots & \square & \square & \square & \ddots & \vdots \\ b_l & \square & \square & \square & \square & \square \end{array}$$

Então, a célula (i, j) da matriz S_1 recebe o valor da equação abaixo e a célula (j, i) recebe zero. Caso contrário, deve-se inserir o valor de 0,5 nas células (i, j) e (j, i) .

$$Value = \frac{\sum_{j=1}^n (t_j | a_i P_j b_k)}{\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k \cup a_i I_j b_k \cup a_i R_j b_k \cup b_k Q_j a_i \cup b_k P_j a_i)}$$

$$S_1: \begin{array}{c|cccccc} & a_i & b_1 & b_2 & \dots & b_l \\ \hline a_i & \square & \square & \square & \dots & \square \\ b_1 & \square & \square & \square & \dots & \square \\ b_2 & \square & \square & \square & \dots & \square \\ \vdots & \square & \square & \square & \ddots & \vdots \\ b_l & \square & \square & \square & \square & \square \end{array}$$

Cenário S_2 :

$$a_i T_{S_2} b_k$$

$$SE \left[\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k) > \sum_{j=1}^n (t_j | a_i I_j b_k \cup a_i R_j b_k \cup b_k Q_j a_i \cup b_k P_j a_i) \right]$$

$$E [(b_{k,j} - a_{i,j}) < d_j, \forall j]$$

Se a alternativa/perfil da linha i dominar a alternativa/perfil da coluna j , deve-se inserir “T” na célula correspondente da matriz a seguir.

$$T_2:$$

	a_i	b_1	b_2	...	b_l
a_i	$\square$	$\square$	$\square$	...	$\square$
b_1	$\square$	$\square$	$\square$	...	$\square$
b_2	$\square$	$\square$	$\square$	...	$\square$
$\vdots$	$\square$	$\square$	$\square$	$\ddots$	$\vdots$
b_l	$\square$	$\square$	$\square$	$\square$	$\square$

A célula (i, j) da matriz S_2 recebe o valor da seguinte equação abaixo e a célula (j, i) recebe zero. Caso contrário, deve-se inserir o valor de 0,5 nas células (i, j) e (j, i) .

$$Value = \frac{\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k)}{\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k \cup a_i I_j b_k \cup a_i R_j b_k \cup b_k Q_j a_i \cup b_k P_j a_i)}$$

$$S_2:$$

	a_i	b_1	b_2	...	b_l
a_i	$\square$	$\square$	$\square$	...	$\square$
b_1	$\square$	$\square$	$\square$	...	$\square$
b_2	$\square$	$\square$	$\square$	...	$\square$
$\vdots$	$\square$	$\square$	$\square$	$\ddots$	$\vdots$
b_l	$\square$	$\square$	$\square$	$\square$	$\square$

Cenário S_3 :

$$a_i T_{S_3} b_k$$

$$SE \left[\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k \cup a_i I_j b_k) > \sum_{j=1}^n (t_j | a_i R_j b_k \cup b_k Q_j a_i \cup b_k P_j a_i) \right]$$

$$E [(b_{k,j} - a_{i,j}) < d_j, \forall j]$$

Se a alternativa/perfil da linha i dominar a alternativa/perfil da coluna j , deve-se inserir “T” na célula correspondente da matriz a seguir.

$$T_3:$$

	a_i	b_1	b_2	...	b_l
a_i	$\square$	$\square$	$\square$	...	$\square$
b_1	$\square$	$\square$	$\square$	...	$\square$
b_2	$\square$	$\square$	$\square$	...	$\square$
$\vdots$	$\square$	$\square$	$\square$	$\ddots$	$\vdots$
b_l	$\square$	$\square$	$\square$	$\square$	$\square$

A célula (i, j) da matriz S_3 recebe o valor da seguinte equação abaixo e a célula (j, i) recebe zero. Caso contrário, deve-se inserir o valor de 0,5 nas células (i, j) e (j, i) .

$$Value = \frac{\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k \cup a_i I_j b_k)}{\sum_{j=1}^n (t_j | a_i P_j b_k \cup a_i Q_j b_k \cup a_i I_j b_k \cup a_i R_j b_k \cup b_k Q_j a_i \cup b_k P_j a_i)}$$

$$S_3: \begin{array}{c} \begin{array}{cccccc} \square & a_i & b_1 & b_2 & \dots & b_l \\ a_i & \square & \square & \square & \dots & \square \\ b_1 & \square & \square & \square & \dots & \square \\ b_2 & \square & \square & \square & \dots & \square \\ \vdots & \square & \square & \square & \ddots & \vdots \\ b_l & \square & \square & \square & \square & \square \end{array} \end{array}$$

2.7 Verificar a consistência dos perfis em relação a cada cenário. Por exemplo, verificar se b_k domina b_{k+1} nos cenários S_1, S_2 e S_3 .

2.8 Para cada cenário, deve-se atribuir a_i a uma das classes ordenadas, com base nas relações de dominância de THOR correspondentes. Duas regras podem ser testadas:

Regra otimista: Começando pelo pior perfil (b_{l-1}), compara-se a_i com cada perfil até ser encontrada a relação $b_k T a_i$. A alternativa é, então, atribuída à classe abaixo do perfil. Se nenhum perfil domina a alternativa a_i , essa alternativa deve ser atribuída à classe C_1 .

Regra pessimista: Começando pelo melhor perfil (b_1), compara-se a_i com cada perfil até ser encontrada a relação $a_i T b_k$. A alternativa é, então, atribuída à classe acima do perfil b_k . Se a alternativa não domina nenhum perfil, então essa alternativa deve ser atribuída à classe C_l .

3º Passo: Para cada cenário específico, dada a classificação de todas as alternativas do 2º Passo, uma classificação mais restritiva pode ser obtida aplicando-se a seguinte regra: Se uma alternativa a_i é dominada por a_j e a_i foi designado para uma classe melhor do que a_j (devido à intransitividade do THOR 2), então, rebaixa-se a_i para a mesma classe que a_j .

4. VALIDAÇÃO DO MÉTODO

4.1. APLICAÇÃO NO THOR 2

No artigo de Esteves *et al.* (2022), foi estudada a priorização de demandas da área de Suprimentos de uma empresa multinacional, por meio da abordagem VFT e do método AMD THOR 2. Para este artigo, foi realizada uma comparação dos resultados encontrados no estudo citado e a aplicação dos dados no método AMD THOR-Sort.

Após a etapa do VFT, foi enviado um questionário para quatro especialistas da área de compras, para atribuição de pesos de 1 a 7 para cada critério, sendo 1 menos importante e 7 mais importante. Dessa forma, foi possível definir o grau de relevância de cada critério. Na Tabela 1 estão os pesos estabelecidos aos critérios pelos compradores, sendo C_1, C_2, C_3 e C_4 , a identificação de cada especialista. Os quatro primeiros critérios são quantitativos e os três últimos, qualitativos (ESTEVES *et al.*, 2022).

Tabela 1: Pesos dos Critérios

Critérios	Pesos dos Critérios			
	C_1	C_2	C_3	C_4
Tempo da requisição	7	5	3	3
Valor da requisição	7	3	6	3
Possíveis ganhos financeiros	7	6	7	4
Lead Time de entrega	7	6	3	5

Risco de parada de planta	7	7	7	7
Risco de impactar os funcionários	7	7	4	5
Risco de impactar o cliente	7	7	5	5

Fonte: ESTEVES *et al.* (2022).

Foram atribuídos, em consenso entre os especialistas da área de compras em questão, os valores para os critérios qualitativos, limites de pertinência (*p*), indiferença (*q*) e discordância (*D*) para os critérios, como mostrado na Tabela 2. Os pesos foram agregados por meio da soma dos pesos normalizados, pelo THOR 2. Na Tabela 3, constam as pertinências consideradas para a atribuição dos valores de cada critério em relação a cada alternativa. As pertinências possuem uma escala de 0 a 1, sendo quanto mais próximo à 1, maior grau de segurança do dado inserido. Para as pertinências dos pesos mostrados na Tabela 1, foi atribuído valor 1. As demandas analisadas são as mais comuns entre as requisições recebidas (ESTEVES *et al.*, 2022).

Tabela 2: Matriz de Decisão

	Tempo da requisição (dias)	Valor da requisição (R\$)	Possíveis ganhos financeiros (R\$)	Lead Time de entrega (dias)	Risco de parada de planta	Risco de impactar os funcionários	Risco de impactar o cliente
Licença de Operação	10	15000	2000	60	5	5	5
Análise do Solo	8	35000	3000	90	2	1	1
Análise das Águas Subterrâneas	9	59450	10500	60	2	1	1
Análise de Ruído Ocupacional	10	7900	50	20	2	4	1
Compra de EPI	11	26676	0	60	2	5	1
Treinamento de Brigada de Incêndio	12	9900	1620	5	1	4	1
Manutenção de Extintores	7	12500	0	25	3	4	2
<i>Pesos</i>	2,57143	2,71428	3,42857	3,0	4,0	3,28572	3,42858
<i>p</i>	4	20000	10000	20	1	1	1
<i>q</i>	1	5000	2000	5	0,5	0,5	0,5
<i>D</i>	5,5	56705	11550	93,5	4,4	4,4	4,4

Fonte: ESTEVES *et al.* (2022).

Tabela 3: Matriz de Pertinências

	Tempo da requisição (dias)	Valor da requisição (R\$)	Possíveis ganhos financeiros (R\$)	Lead Time de entrega (dias)	Risco de parada de planta	Risco de impactar os funcionários	Risco de impactar o cliente
Licença de Operação	1	1	0,7	1	0,8	0,8	0,8
Análise do Solo	1	1	0,7	1	0,6	0,6	0,6
Análise das Águas Subterrâneas	1	1	0,7	1	0,6	0,6	0,6
Análise de Ruído Ocupacional	1	1	0,7	1	0,6	0,6	0,6

Compra de EPI	1	1	0,7	1	0,6	0,6	0,6
Treinamento de Brigada de Incêndio	1	1	0,7	1	0,7	0,7	0,7
Manutenção de Extintores	1	1	0,7	1	0,7	0,7	0,7

Fonte: ESTEVES *et al.* (2022).

Após isso, foram realizados os cálculos por meio do THOR 2, e as alternativas foram ordenadas nos cenários S1, S2 e S3. Na Tabela 4 estão os resultados obtidos.

Tabela 4: Ordenação das alternativas nos cenários S1, S2 e S3

S1		S2		S3	
Licença de Operação	3,193	Licença de Operação	4,092	Licença de Operação	5,105
Compra de EPI	3,0	Compra de EPI	3,426	Compra de EPI	3,373
Análise do Solo	3,0	Análise de Águas Subterrâneas	2,777	Análise de Águas Subterrâneas	3,596
Análise de Águas Subterrâneas	2,569	Análise do Solo	2,544	Análise do Solo	3,192
Manutenção de Extintores	2,5	Manutenção de Extintores	1,575	Manutenção de Extintores	2,241
Treinamento de Brigada de Incêndio	2,5	Análise de Ruído Ocupacional	1,0	Análise de Ruído Ocupacional	1,437
Análise de Ruído Ocupacional	2,0	Treinamento de Brigada de Incêndio	0,5	Treinamento de Brigada de Incêndio	0,558

Fonte: ESTEVES *et al.* (2022).

4.2. APLICAÇÃO NO THOR-SORT

Os dados do problema supracitado foram inseridos no código computacional do THOR-Sort a fim de que os resultados fossem comparados.

Além dos dados pré-estabelecidos, foi necessário definir as três classes do problema: alta, média e baixa prioridades, sendo 1, 2 e 3, respectivamente. E, também, foram definidos os perfis de fronteira (B1 e B2), que ficam entre as classes (Tabela 5).

Tabela 5: Perfis de Fronteira

	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇
B1	11	30000	4000	50	4	4	4
B2	9	20000	3000	30	2	2	2

Fonte: Autores (2024).

Sendo, A₁ – Licença de Operação; A₂ – Análise do Solo; A₃ – Análise de Águas Subterrâneas; A₄ – Análise de Ruído Ocupacional; A₅ – Compra de EPI; A₆ – Treinamento de Brigada de Incêndio; e A₇ – Manutenção de Extintores.

Os resultados obtidos na classificação pela regra otimista e pela regra pessimista estão demonstrados nas Tabelas 6 e 7, respectivamente, considerando os cenários S1, S2 e S3:

Tabela 6: Classificação Otimista

OTIMISTA	S1	S2	S3
Licença de Operação	1	1	1
Análise do Solo	2	2	3
Análise de Águas Subterrâneas	2	2	3
Análise de Ruído Ocupacional	2	3	3

Compra de EPI	1	2	3
Treinamento de Brigada de Incêndio	2	3	3
Manutenção de Extintores	1	2	3

Fonte: Autores (2024).

Tabela 7: Classificação Pessimista

PESSIMISTA	S1	S2	S3
Licença de Operação	2	1	1
Análise do Solo	3	3	2
Análise de Águas Subterrâneas	3	3	2
Análise de Ruído Ocupacional	3	3	3
Compra de EPI	3	3	2
Treinamento de Brigada de Incêndio	3	3	3
Manutenção de Extintores	3	3	2

Fonte: Autores (2024).

4.3. COMPARAÇÃO DOS RESULTADOS

Verifica-se que a alternativa Licença de Operação, que ficou em primeiro lugar na ordenação do THOR 2, manteve-se na prioridade alta em todos os cenários da classificação otimista e nos cenários S2 e S3 da classificação pessimista. A Compra de EPI, que havia ficado em segundo lugar ao utilizar-se o método THOR 2, se estabeleceu na classe de prioridade alta no cenário S1 da classificação otimista. Já as alternativas Treinamento de Brigada de Incêndio e Análise de Ruído Ocupacional, que estavam por último na ordenação do THOR 2, ocuparam a classe de prioridade baixa no cenário S3 da classificação pessimista, ao passo de que as alternativas Análise do Solo, Análise de Águas Subterrâneas, Compra de EPI e Manutenção de Extintores, ficaram na média prioridade nesse mesmo cenário.

A alternativa Manutenção de Extintores ficou em prioridade alta no cenário S1 da classificação otimista, ao passo em que ocupava a quarta colocação na ordenação do THOR 2. Isso pode ter ocorrido devido à regra de sobreclassificação em S1 ser mais exigente e, assim, no cenário otimista, nenhum perfil foi capaz de superar essa alternativa pela regra especificada. Nota-se que, na regra otimista, uma alternativa passa para a classe acima do perfil a menos que este a supere pela regra do cenário em questão. Além disso, há diferenças no algoritmo do THOR-Sort em comparação ao THOR 2, principalmente por apresentar as classificações otimista e pessimista, que não existem no método THOR 2.

Também se nota que o cenário S1 da classificação otimista e o S3 da pessimista são os que mais se aproximam entre si e refletem a realidade. Para isso, há uma explicação lógica: ao passo que os cenários ficam mais flexíveis, pela regra de classificação otimista, determinado perfil poderia superar uma alternativa que, assim, seria alocada em uma classe menos preferível. Já na regra pessimista, que é uma classificação mais exigente, inicia-se em um cenário de mais alternativas na prioridade baixa e, à medida que fica mais flexível, mais alternativas aparecem nas prioridades média e alta. Isso ocorre porque, na regra pessimista, a alternativa que deve superar o perfil. Isso acaba acontecendo com maior frequência em cenários mais flexíveis.

De forma geral, foi possível demonstrar que o método AMD THOR-Sort foi capaz de classificar as alternativas de forma consistente com os passos do procedimento, os resultados do THOR 2, e as expectativas do seu uso nesta aplicação numérica de validação.

5. TUTORIAL DO SOFTWARE DO THOR-SORT

O código computacional do THOR-Sort pode ser executado em uma ferramenta como o *Jupyter Notebook*, conforme as etapas a seguir.

Inicialmente importa-se a biblioteca NumPy (Figura 1), que fornece suporte para *arrays* multidimensionais (como *arrays* ou matrizes) e uma variedade de funções matemáticas de alto nível para operar nesses *arrays*.

THOR-Sort

In this notebook, a THOR-based approach for sorting problems is tested in a numerical application.

Testing a THOR-Sort approach

```
In [1]: import numpy as np
```

Figura 1: Importando biblioteca NumPy.

Fonte: Autores (2024)

Em seguida, na matriz A, deve-se inserir os valores de cada alternativa diante dos critérios, em ordem, como demonstrado na Figura 2:

```
In [2]: A = np.array ([[10,15000,2000,60,5,5,5],[8,35000,3000,90,2,1,1],
                        [9,59450,10500,60,2,1,1],[10,7900,50,20,2,4,1],
                        [11,26676,0,60,2,5,1],[12,9900,1620,5,1,4,1],
                        [7,12500,0,25,3,4,2]])
A
Out[2]: array([[ 10, 15000,  2000,   60,    5,    5,    5],
               [  8, 35000,  3000,   90,    2,    1,    1],
               [  9, 59450, 10500,   60,    2,    1,    1],
               [ 10,  7900,   50,   20,    2,    4,    1],
               [ 11, 26676,    0,   60,    2,    5,    1],
               [ 12,  9900,  1620,    5,    1,    4,    1],
               [  7, 12500,    0,   25,    3,    4,    2]])
```

Figura 2: Matriz de alternativas.

Fonte: Autores (2024)

Após isso, são inseridos os valores de pertinência dos dados das respectivas alternativas, conforme Figura 3:

```
In [3]: A_pert = np.array([[1,1,0.7,1,0.8,0.8,0.8],[1,1,0.7,1,0.6,0.6,0.6],
                           [1,1,0.7,1,0.6,0.6,0.6],[1,1,0.7,1,0.6,0.6,0.6],
                           [1,1,0.7,1,0.6,0.6,0.6],[1,1,0.7,1,0.7,0.7,0.7],
                           [1,1,0.7,1,0.7,0.7,0.7]])
A_pert
Out[3]: array([[1. , 1. , 0.7, 1. , 0.8, 0.8, 0.8],
               [1. , 1. , 0.7, 1. , 0.6, 0.6, 0.6],
               [1. , 1. , 0.7, 1. , 0.6, 0.6, 0.6],
               [1. , 1. , 0.7, 1. , 0.6, 0.6, 0.6],
               [1. , 1. , 0.7, 1. , 0.6, 0.6, 0.6],
               [1. , 1. , 0.7, 1. , 0.7, 0.7, 0.7],
               [1. , 1. , 0.7, 1. , 0.7, 0.7, 0.7]])
```

Figura 3: Matriz de pertinências das alternativas.

Fonte: Autores (2024)

Os perfis de fronteira, são inseridos de acordo com a Figura 4, na matriz B:

```
In [4]: B = np.array([[11,30000,4000,50,4,4,4],[9,30000,3000,30,2,2,2]])
B
Out[4]: array([[ 11, 30000,  4000,   50,    4,    4,    4],
               [  9, 30000,  3000,   30,    2,    2,    2]])
```

Figura 4: Perfis de fronteira.

Fonte: Autores (2024)

Neste caso, como as alternativas são classificadas em três classes, existem dois perfis de fronteira, com valores correspondentes a cada critério.

Os perfis de fronteira também possuem valores de pertinência (neste caso, todos 1), que são incluídos no código conforme Figura 5:

```
In [5]: B_pert = np.ones(shape = B.shape, dtype=int)
        B_pert

Out[5]: array([[1, 1, 1, 1, 1, 1, 1],
               [1, 1, 1, 1, 1, 1, 1]])
```

Figura 5: Pertinências dos perfis de fronteira.

Fonte: Autores (2024)

Em seguida, conforme Figura 6, devem ser inseridos os valores de indiferença (q), preferência (p), discordância (d), bem como os pesos dos critérios (w) e suas pertinências (w_pert):

```
In [6]: q = np.array([1,5000,2000,5,0.5,0.5,0.5])
        p = np.array([4,20000,10000,20,1,1,1])
        d = np.array([5.5,56705,11550,93.5,4.4,4.4,4.4])
        w = np.array([2.57143,2.71428,3.42857,3,4,3.28572,3.42858])
        w_pert = np.array([1,1,1,1,1,1,1])
```

Figura 6: Valores de indiferença, preferência, discordância, pesos e pertinência dos pesos.

Fonte: Autores (2024)

Importa-se o THOR-Sort, como mostrado na Figura 7:

```
In [7]: %load_ext autoreload
        %autoreload 2

        from ThorSort import ThorSort
```

Figura 7: Importando o THOR-Sort.

Fonte: Autores (2024)

São definidas as situações otimista e pessimista do THOR-Sort, de acordo com os dados inseridos e os parâmetros estabelecidos, conforme Figura 8:

```
In [8]: opt, pes = ThorSort (DecisionMatrix = A, DecisionMatrix_pert = A_pert,
                             Profiles = B, Profiles_pert = B_pert,
                             p = p, q = q, d = d, w = w, w_pert = w_pert, post_class= False)
```

Figura 8: Definições das situações otimista e pessimista.

Fonte: Autores (2024)

Ao rodar todas as linhas do código, são fornecidos os resultados da situação otimista e pessimista (Figura 9):

In [9]:	opt	In [10]:	pes																																																																
out[9]:	<table> <tr><th></th><th>S1</th><th>S2</th><th>S3</th></tr> <tr><td>a_1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>a_2</td><td>2</td><td>2</td><td>3</td></tr> <tr><td>a_3</td><td>2</td><td>2</td><td>3</td></tr> <tr><td>a_4</td><td>2</td><td>3</td><td>3</td></tr> <tr><td>a_5</td><td>1</td><td>2</td><td>3</td></tr> <tr><td>a_6</td><td>2</td><td>3</td><td>3</td></tr> <tr><td>a_7</td><td>1</td><td>2</td><td>3</td></tr> </table>		S1	S2	S3	a_1	1	1	1	a_2	2	2	3	a_3	2	2	3	a_4	2	3	3	a_5	1	2	3	a_6	2	3	3	a_7	1	2	3	out[10]:	<table> <tr><th></th><th>S1</th><th>S2</th><th>S3</th></tr> <tr><td>a_1</td><td>2</td><td>1</td><td>1</td></tr> <tr><td>a_2</td><td>3</td><td>3</td><td>2</td></tr> <tr><td>a_3</td><td>3</td><td>3</td><td>2</td></tr> <tr><td>a_4</td><td>3</td><td>3</td><td>3</td></tr> <tr><td>a_5</td><td>3</td><td>3</td><td>2</td></tr> <tr><td>a_6</td><td>3</td><td>3</td><td>3</td></tr> <tr><td>a_7</td><td>3</td><td>3</td><td>2</td></tr> </table>		S1	S2	S3	a_1	2	1	1	a_2	3	3	2	a_3	3	3	2	a_4	3	3	3	a_5	3	3	2	a_6	3	3	3	a_7	3	3	2
	S1	S2	S3																																																																
a_1	1	1	1																																																																
a_2	2	2	3																																																																
a_3	2	2	3																																																																
a_4	2	3	3																																																																
a_5	1	2	3																																																																
a_6	2	3	3																																																																
a_7	1	2	3																																																																
	S1	S2	S3																																																																
a_1	2	1	1																																																																
a_2	3	3	2																																																																
a_3	3	3	2																																																																
a_4	3	3	3																																																																
a_5	3	3	2																																																																
a_6	3	3	3																																																																
a_7	3	3	2																																																																

Figura 9: Resultados.

Fonte: Autores (2024)

A ordem das alternativas segue conforme a inserção dos dados nas etapas anteriores.

Após utilizar o THOR-Sort para obtenção dos resultados, é necessário que o decisor os interprete e realize as análises necessárias diante da decisão a ser tomada.

6. CONCLUSÕES

Diante dos resultados obtidos com a utilização do THOR-Sort, é possível atestar que o THOR-Sort é um método efetivo para classificação de alternativas, fornecendo uma assertividade, confiabilidade e consistência nos resultados. Na validação do método, foi possível obter a classificação de demandas para um estudo que, anteriormente, teve suas alternativas ordenadas. Isso foi de grande valia, pois para esse tipo de problemática, a classificação das alternativas é o resultado mais apropriado. E, em paralelo, como os resultados encontrados foram bem fidedignos aos anteriores, foi possível demonstrar a aderência do novo método AMD THOR-Sort. O tutorial do *software* demonstrado neste artigo é de suma importância para contribuir com a comunidade acadêmica e com organizações públicas e privadas, visto que é possível visualizar a utilização do THOR-Sort de forma exemplificada, possibilitando a expansão do conhecimento.

7. REFERÊNCIAS

- BRANS, J. P.; VINCKE, P. H.** A preference ranking organization method, the PROMETHEE method for MCDM. *Management Science*, [s.l.], v. 31, n. 6, p. 647–656, 1985.
- COSTA, C. A. B.; VANSNICK, J. C.** MACBETH - An Interactive Path Towards the Construction of Cardinal Value Functions. *International Transactions in Operational Research*, Nova Jersey, v. 1, n. 4, p. 489–500, 1994.
- EDWARDS, W.** How to Use Multiattribute Utility Measurement for Social Decision Making. *IEEE Transactions on Systems, Man, and Cybernetics*, [s.l.], v. 7, n. 5, p. 326–340, 1977.
- ESTEVES, F. M. S.; MOREIRA, M. A. L.; COSTA, I. P. A.; JOAQUIM, G. P.; GOMES, C. F. S.; DOS SANTOS, M.** Demand Prioritization on Supply Chain by the Integration of Value-Focused Thinking Approach and THOR 2 Method. *Procedia Computer Science*, Manchester, v. 214, p. 248-256, 2022.
- ESTEVES, F. M. S.** THOR-SORT: Um Novo Método Multicritério para Classificação de Alternativas. Dissertação (Mestrado em Engenharia de Produção) – Universidade Federal Fluminense, Programa de Pós-graduação em Engenharia de Produção. Niterói, 2024.
- GOMES, C. F. S.; DOS SANTOS, M.; TEIXEIRA, L. F. H.; SANSEVERINO, A. M.; BARCELOS, M. R. D. S.** SAPEVO-M: a group multicriteria ordinal ranking method. *Pesquisa Operacional*, Rio de Janeiro, v. 40, p. 1-23, 2020.

GOMES, C. F. S.; NUNES, K. R. A.; XAVIER, L. H.; CARDOSO, R.; VALLE, R. Multicriteria decision making applied to waste recycling in Brazil. *Production*, Rio de Janeiro, v. 36, n. 3, p. 395–404, 2008.

GOMES, C. F. S. THOR - Um Algoritmo Híbrido de Apoio Multicritério à Decisão para Processos Decisórios com Alternativas Discretas. 241 p. Tese (Doutorado em Engenharia de Produção) — Universidade Federal do Rio de Janeiro, Pós-Graduação e Pesquisa de Engenharia. Rio de Janeiro, 1999.

GOMES, L. F. A. M.; ARAYA, M. C. G.; CARIGNANO, C. Tomada de Decisões em Cenários Complexos. São Paulo: CENGAGE, 2003.

GOMES, L. F. A. M.; LIMA, M. M. P. P. TODIM: Basics and application to multicriteria ranking of projects with environmental impacts. *Foundations of Computing and Decision Sciences*, Poznan, v. 16, n. 4, 113–127, 1991.

HWANG, C. L.; YOON, K. Multiple Attribute Decision Making. Berlin: Springer Berlin Heidelberg, 1981.

INFANTE, R. P.; DE OLIVEIRA, M. C.; ASSUMPÇÃO, M. R. P. Uso de métodos multicritérios de tomada de decisão para seleção dos objetivos estratégicos e indicadores de performance considerados no Balanced Scorecard. *Revista de Ciência & Tecnologia, Boavista*, v. 20, n. 39, p. 45-58, 2017.

RANGEL, L. A. D.; GOMES, L. F. A. M. O Apoio Multicritério à Decisão na avaliação de candidatos. *Production*, Rio de Janeiro, v. 20, n. 1, p. 92–101, 2010.

ROY, B. Classement et choix en présence de points de vue multiples. *Revue française d'informatique et de recherche opérationnelle*, EDP Sciences, [s.l.], v. 2, n. 8, p. 57–75, 1968.

SAATY, T. L. The Analytic Hierarchy Process. New York: McGraw-Hill, 1980.

SAATY, T. L.; VARGAS, L. G. Decision making with the analytic network process: economic, political, social and technological applications with benefits, opportunities, costs and risks. 95. ed. New York: Springer, 2006.

SOBRAPO. (2022). O que é Pesquisa Operacional? Sociedade Brasileira de Pesquisa Operacional. Disponível em: <<https://www.sobrapo.org.br/o-que-e-pesquisa-operacional>>. Acesso em: 07 de março de 2024.

SILVA, M. D. C., GOMES, C. F. S.; DA COSTA, C. L. A hybrid multicriteria methodology topsis-macbeth-2n applied in the ordering of technology transfer offices. *Pesquisa Operacional*, v. 38, n. 3, 2018.

TENÓRIO, F. M.; DOS SANTOS, M.; GOMES, C. F. S. Revisitando o método THOR: uma pesquisa bibliométrica. XIX Simpósio De Pesquisa Operacional e Logística da Marinha, Rio de Janeiro, p. 1-14, 06 a 08 nov. 2019.

TENÓRIO, F. M. Modelagem multicritério: uma evolução do método THOR. 2020. 117 p. Dissertação (Mestrado em Sistemas e Computação) – Instituto Militar de Engenharia, Programa de Pós-Graduação em Sistemas e Computação. Rio de Janeiro, 2020.

KEENEY, R. L.; RAIFFA, H. Decisions with multiple objectives: preferences and value tradeoffs. 2. ed. Cambridge: Cambridge University Press, 1993.