МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ЧИСЛЕННЫЕ МЕТОДЫ И КОМПЛЕКСЫ ПРОГРАММ / MATHEMATICAL MODELING, NUMERICAL METHODS AND PROGRAM COMPLEXES

ENHANCING MULTI-CRITERIA DECISION-MAKING: A NOVEL APPROACH TO GENERALIZED DECISION RULES USING MAHALANOBIS DISTANCE

Research article

Hammoud A.[1], *
[1] ORCID : 0009-0004-6063-0327;
[1] Kuban State Agrarian University, Krasnodar, Russian Federation

* Corresponding author (ali-hammoud[at]mail.ru)

Abstract

This paper introduces the Modified Generalized Decision Rules (MaGDR) algorithm, an enhancement of the traditional Generalized Decision Rules (GDR) algorithm, which integrates the Mahalanobis distance to address inherent limitations in multi-criteria decision-making (MCDM). Traditional GDR algorithms often struggle with correlation and scaling issues among different criteria, leading to suboptimal decision-making. By incorporating the Mahalanobis distance, the MaGDR algorithm provides a robust, scale-invariant method that accurately reflects true distances within the criteria space, thus improving decision-making accuracy and reliability. We detail the theoretical modifications to the GDR algorithm, emphasizing the computational mechanics and integration of the Mahalanobis distance. The implementation is demonstrated through a simulation of robotic pathfinding, a complex decision-making scenario characterized by obstacle avoidance. The simulation results validate the MaGDR algorithm's enhanced capability to handle intricate pathfinding tasks, showing improvements in decision-making efficiency compared to the Dijkstra algorithm. The findings suggest that the MaGDR algorithm not only performs effectively in typical MCDM applications but also excels in scenarios requiring nuanced assessments of multiple conflicting criteria. Future researches will explore the scalability of this approach in other domains, such as healthcare and logistics, and seek to optimize iteration efficiency further.

Keywords: Multi-Criteria Decision-Making (MCDM), Generalized Decision Rules (GDR), MaGDR, Mahalanobis distance, decision metrics.

УЛУЧШЕНИЕ ПРОЦЕССА ПРИНЯТИЯ МНОГОКРИТЕРИАЛЬНЫХ РЕШЕНИЙ: НОВЫЙ ПОДХОД К ОБОБЩЕННЫМ РЕШАЮЩИМ ПРАВИЛАМ С ИСПОЛЬЗОВАНИЕМ РАССТОЯНИЯ МАХАЛАНОБИСА

Научная статья

Хаммуд А.[1], *
[1] ORCID : 0009-0004-6063-0327;
[1] Кубанский государственный аграрный университет, Краснодар, Российская Федерация

* Корреспондирующий автор (ali-hammoud[at]mail.ru)

Аннотация

В данной статье представлен алгоритм модифицированных обобщенных правил принятия решений (MaGDR), усовершенствование алгоритма традиционных обобщенных правил принятия решений (GDR), который объединяет расстояние Махаланобиса для устранения присущих ограничений при принятии многокритериальных решений (MCDM). Традиционные алгоритмы GDR часто сталкиваются с проблемами корреляции и масштабирования между различными критериями, что приводит к принятию неоптимальных решений. Включив расстояние Махаланобиса, алгоритм MaGDR обеспечивает надежный, масштабно-инвариантный метод, который точно отражает истинные расстояния в пространстве критериев, тем самым повышая точность и надежность принятия решений. Мы подробно описываем теоретические модификации алгоритма GDR, уделяя особое внимание вычислительной механике и интеграции расстояния Махаланобиса. Реализация демонстрируется посредством моделирования роботизированного поиска пути — сложного сценария принятия решений, характеризующегося обходом препятствий. Результаты моделирования подтверждают расширенные возможности алгоритма MaGDR решать сложные задачи поиска пути, демонстрируя повышение эффективности принятия решений по сравнению с алгоритмом Dijkstra. Результаты показывают, что алгоритм MaGDR не только эффективно работает в типичных приложениях MCDM, но также превосходен в сценариях, требующих детальной оценки нескольких конфликтующих критериев. Будущие исследования будут изучать масштабируемость этого подхода в других областях, таких как здравоохранение и логистика, и стремиться к дальнейшей оптимизации эффективности итераций.

Ключевые слова: многокритериальное принятие решений (MCDM), обобщенное решающее правило (GDR), MaGDR, расстояние Махаланобиса, метрики принятия решений.

Introduction

Multi-Criteria Decision-Making (MCDM) is considered as a field in operations research, management science, and multiagent systems, addressing problems where decisions must be made in the presence of trade-offs between two or more conflicting criteria. In various fields such as engineering, finance, healthcare, and environmental planning, MCDM facilitates the evaluation and comparison of different options based on a set of criteria rather than a single criterion [1], [2], [3]. For example, in autonomous vehicles navigation system is modelled as a multiagent system, MCDM is used to balance trade-offs

between minimizing travel time, reducing energy consumption, and maximizing passenger safety. All of them are influenced by different environmental and traffic conditions. One of the critical challenges in MCDM is to accurately synthesize and prioritize diverse sets of criteria to make a decision that aligns with strategic objectives and operational frameworks.

The Generalized Decision Rules (GDR) algorithm is a robust tool within MCDM techniques, particularly valuable in complex decision support systems. GDR applies a set of rules that systematically evaluate each option against all criteria. These rules typically involve aggregating individual criteria scores through various metrics, such as calculating sums, maximums, and distances to the ideal point, which are used to rank or select the best alternatives [4]. Although, the GDR is effective in many scenarios, it often fails to account for the correlation and relative scaling among criteria. That can lead to suboptimal decision-making when cope with criteria that are interdependent or varying units and scales.

Recognizing the limitations inherent in the traditional GDR algorithm, there is a need to enhance its methodology to incorporate more sophisticated measures that respect the statistical properties of the criteria space. To address these challenges, an integration of the Mahalanobis distance concept into GDR represents a significant advancement in this direction. The Mahalanobis distance, unlike simpler distance measures, considers the covariance among the criteria, so providing a more accurate reflection of the true distances within the criteria space, factoring in correlations and variances that are critical in many decision-making contexts [5]. By considering the Mahalanobis distance concept, the modified GDR algorithm not only respects the multidimensional nature of the decision space but also adjusts for any disproportionate influence of widely varying scales among the criteria.

This paper introduces a novel adaptation of the GDR algorithm that leverages the Mahalanobis distance to improve decision-making accuracy and reliability. That addresses the limitations of traditional GDR by enhancing its capability to process complex, multidimensional space in a way that is both scale-invariant and sensitive to the inherent correlations among decision criteria. This modified approach promises to refine decision support systems by offering a method that is not only theoretically sound, but also practically more effective in dealing with multi-criteria decision problems.

The following sections details the theoretical background of this modified algorithm, describe the methodological innovations, present empirical evidence by applying it to a pathfinding problem in robotics to validate the approach and discuss the potential implications of the findings.

Methodology

This section outlines the foundational principles of the GDR algorithm, the integration of the Mahalanobis distance, and the solutions to computational issues like covariance matrix singularity.

2.1. The Original GDR Algorithm

The GDR algorithm utilizes a structured decision-making data, defined by a set of options $X = \{x_1, x_2, \ldots, x_I\}$ and criteria $C = \{c_1, c_2, \ldots, c_J\}$ [6]. The necessary conditions that are required for effective GDR algorithm:

1. Homogeneity of Criteria.

Within the vector objective function (VTF), all criteria must uniformly adopt an extremum type, either minimized or maximized, to ensure consistency. The selection of this extremum type is important as it determines the metrics subsequently used within the algorithm. The mathematical representation of this concept is shown in Equation 1 and 2.

$$F(x) = \big(F_1(x), F_2(x), \ldots, F_J(x)\big) \tag{1}$$

$$F_v(x) \rightarrow \text{ext}, v = 1, 2, \ldots, J; \ \text{ext} \in \{\min, \max\} \tag{2}$$

If the VTF includes a mix of maximized and minimized criteria. Then modifications are necessary to standardize the extremum type across all criteria. For example, if the goal is to convert the criterion extremum from minimum to the form of maximized ones. There are two main approaches to achieve that:

One approach involves calculating inverse values, specifically by raising the initial minimized values to the power of (-1).

Another approach is to determine some suitable constant $C_v > \max F_v(x)$ and replace the minimized criterion under consideration with the maximized criterion $F'_v(x) = C_v - F_v(x)$

2. Commensurability of Criteria: All criteria should have the same unit of measurement.

3. Scaling: All criteria must be normalized to the same scale, allowing for direct comparisons.

2.2. GDR algorithm metrics and steps

The following steps is used to prepare the dataset for the calculation of decision metrics. Upon establishing the set of options $X = \{x_1, x_2, \ldots, x_I\}$ and criteria $C = \{c_1, c_2, \ldots, c_J\}$, the next phase involves normalizing the criteria using Equation 3 to ensure a uniform scale, followed by applying weights *W* as in Equation 4 to adjust the influence of each criterion according to its perceived importance.

$$x'_{ij} = \frac{x_{ij}}{x_{\max j} - x_{\min j\, j}}; i = 1, 2, \ldots, I; j = 1, 2, \ldots, J \tag{3}$$

$$x''_{ij} = w_j * x'_{ij}; i = 1, 2, \ldots, I; j = 1, 2, \ldots, J \tag{4}$$

2.3. Decision Metrics Explanation

This step includes calculating three primary metrics for each option to facilitate evaluation and comparison:

- SUM/MINSUM or SUM/MAXSUM: This metric sums all criteria for an option, normalized against the minimum or maximum sum across all options, respectively in Equation 5 and 6.

$$\text{SUM/MINSUM}_i = \frac{\sum_j x''_{ij}}{\min_i \sum_j x''_{ij}}; F_v(x) \rightarrow \min \tag{5}$$

$$\text{SUM} \quad \text{MAXSUM}_{i} = \frac{\sum_j x''_{ij}}{\max_i \sum_j x''_{ij}}; F_v(x) \rightarrow \max \tag{6}$$

- MAX/MINMAX or MIN/MAXMIN: The maximum value among criteria for an option, normalized by the minimum maximum across options. Or the minimum value among criteria for an option, normalized by the maximum minimum across options. Respectively in Equation 7 and 8.

$$\text{MAX/MINMAX}_{i} = \frac{\max''_{j\ ij}}{\min_i \max_j x''_{ij}}; F_v(x) \rightarrow \min \tag{7}$$

$$\text{MIN MAXMIN}_i = \frac{\min_j x''_{ij}}{\max_i \min_j x''_{ij}}; F_v(x) \to \max \tag{8}$$

- DTI/MINDTI or DTI/MAXDTI: Measures the Euclidean distance to an ideal point, normalized by the minimum or maximum distance across options, respectively, offering a sense of how each option deviates from an idealized points as in Equation 9 and 10.

$$\text{DTI/MINDTI } i_i = \frac{\sqrt{\sum_j \left(\min_i x''_j - x''_{ij}\right)^2}}{\min_i \sqrt{\sum_j \left(\min_i x''_j - x''_{ij}\right)^2}}; F_v(x) \to \min \tag{9}$$

$$D_{TI} \; \text{MAXDTI} \;_{i} = \frac{\sqrt{\sum_{j}\left(\max_{i} x''_{j} - x_{ij}\right)^{2}}}{\max_{i} \sqrt{\sum_{j}\left(\max_{i} x''_{j} - x_{ij}\right)^{2}}} \;; F_{v}(x) \rightarrow \max \tag{10}$$

2.4. Iterative Selection

Upon calculating the metrics, a set of options alongside a corresponding set of metrics is formulated. The GDR algorithm utilizes an iterative selection process to streamline decision-making (See figure 1). In this process, each option is evaluated against established metrics. If an option's normalized metrics reach a value of 1 as, indicating optimal alignment with the decision criteria, that option is selected and removed from the original set of options and criteria. Then the process restarts, by calculating metrics for the remaining options. This cyclical evaluation continues until all options are systematically assessed

and selected based on their metrics. As a result, this approach orders the options, ensuring a structured and gradual selection of options.
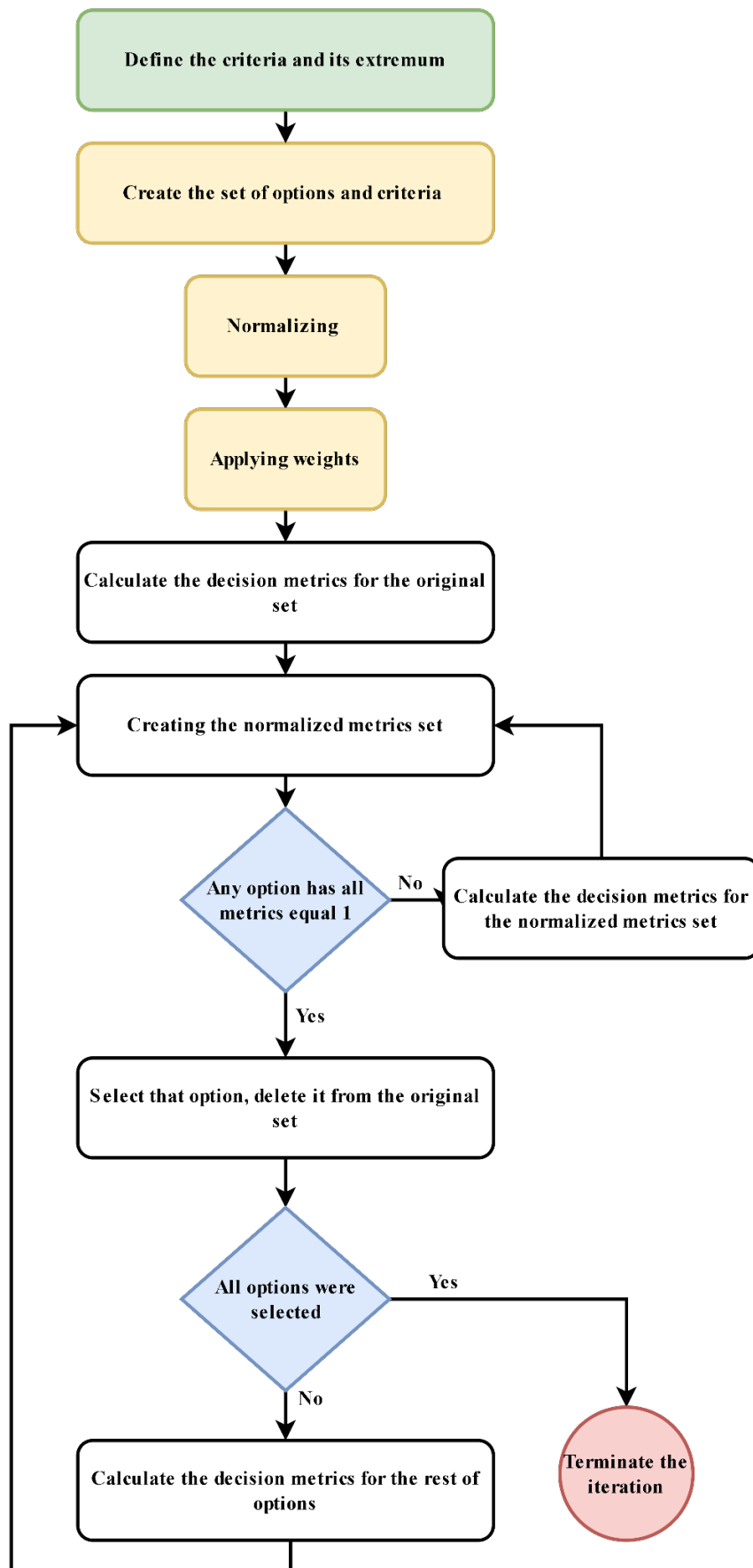

Figure 1 - GDR flowchart
DOI: https://doi.org/10.60797/COMP.2024.3.2.1

2.5. Proposed Modifications

In general, GDR algorithm is robust, although it presents certain limitations that can reduce its reliability in complex decision-making scenarios. Notably, these limitations include issues such as the potential for division by zero in distance calculations, the lack of uniformity in unit measurement across criteria, and the consequent need for normalization. These limitations highlight the necessity for methodological enhancements to improve the algorithm's adaptability and accuracy.

2.6. Addressing Division by Zero in GDR

A significant computational challenge within the GDR occurs when calculating the normalized distances, particularly when the minimum distance metric equals zero, leading to division by zero errors. This issue arises in scenarios where an option perfectly matches the ideal or reference point, making the denominator in the DTI/MINDTI normalized metric calculation zero. To circumvent this problem, our modification introduces a break in the iteration process whenever this condition is encountered, automatically selecting that option as the optimal choice. This approach not only resolves the division by zero issue but also streamlines the decision-making process by promptly identifying the most suitable option.

2.7. Incorporating Mahalanobis Distance into GDR

The Mahalanobis distance offers a more sophisticated alternative to the traditional Euclidean distance. Unlike Euclidean distance, which measures straight-line distance irrespective of data distribution, the Mahalanobis distance takes into account the variance-covariance structure of the data. The use of Mahalanobis distance does not necessitate the normalization or standardization of data across different units of measurement, as it inherently adjusts for scale differences and correlations through its reliance on the covariance matrix. This makes it particularly useful in multivariate settings where variables may not be on the same scale. Thus, it provides a robust, scale-invariant method for measuring distances that reflect the underlying relationships within the data.

$$d_{\mathrm{Mahalanobis}} = \sqrt{\left(x_{ij} - x_{\mathrm{ideal}}\right)^T * C^{-1} * \left(x_{ij} - x_{\mathrm{ideal}}\right)} \tag{11}$$

Where:

x_{ij} : The value of option for criterion j.

x_{ideal} : The ideal point (Minimum point/Maximum point)

C: The covariance matrix.

2.8. Handling Covariance Matrix Singularity

The issue in calculating Mahalanobis distance is the singularity of the covariance matrix, particularly when the data points are collinear or the number of criteria exceeds the number of options. A singular matrix is non-invertible, which prevent the computation of Mahalanobis distance. To tackle this, the solution involves employing the pseudoinverse of the covariance matrix rather than its traditional inverse. The pseudoinverse, generally calculated using methods such as the Moore-Penrose algorithm, offers a robust alternative that effectively handles cases where the matrix may not be directly invertible [7].

2.9. Implementation of the Modified GDR (MaGDR) Algorithm

The implementation of MaGDR algorithm incorporates the Mahalanobis distance to enhance the decision-making process, specifically designed to overcome the limitations identified in traditional GDR.

Setup and Steps

1. Data Collection and Pre-processing:

- Gather all relevant data required for the decision-making process, ensuring comprehensive coverage of options and criteria.

- Define the set of options $X = \{x_1, x_2, \ldots, x_I\}$ and criteria $C = \{c_1, c_2, \ldots, c_J\}$

- Optionally Apply weights *W* as in Equation 12, omitting normalization to maintain the original units of measurement.

$$x'_{ij} = w_j * x_{ij}; i = 1, 2, \ldots, I; j = 1, 2, \ldots, J \tag{12}$$

2. Computation of Decision Metrics:
   - SUM/MINSUM or SUM/MAXSUM
   - MAX/MINMAX or MIN/MAXMIN
   - DTI/MINDTI or DTI/MAXDTI

Applying Mahalanobis distance to calculate DTI metric:

- Calculate the covariance matrix for the criteria or metrics set. This matrix will reflect the variance and covariance.

- In cases where the covariance matrix is singular, employ the pseudoinverse technique to preserve its usability for the calculations.

- For each option, compute the Mahalanobis distance as in Equation 11 from the option to the ideal point. This step requires the inversion or pseudoinversion of the covariance matrix that calculated previously.

3. Iterative Selection Process:

- Apply the iterative selection process where option that meet the condition of all its metrics equaling 1 is selected.

- This iterative process systematically refines the set of options, methodically filtering from the most to the least suitable choices.

2.10. Application of MaGDR in Robotics Pathfinding

Pathfinding in robotics fundamentally involves complex decision-making, where the goal is to determine the best route from a starting point to a destination in an environment with potential obstacles. Efficient and safe navigation is critical in various robotics sectors, such as autonomous vehicles, automated warehousing, and rescue operations. Robotic pathfinding reflects MCDM challenges, including dynamic obstacle avoidance, travel distance optimization, energy consumption

minimization, and strict adherence to safety protocols. Each route must be evaluated on multiple criteria that often conflict [8]. The MaGDR algorithm offers a sophisticated method to address this multi-dimensional evaluation. The primary objective of this simulation is to verify and validate the MaGDR algorithm's decision-making capabilities in complex environments.

2.11. Simulation Setup

The simulation of the MaGDR and Dijkstra algorithms [9] is conducted within a controlled virtual environment. The primary tools utilized include the Python programming language [10], supplemented by essential libraries such as Matplotlib for visualization, NumPy for numerical operations, and pandas for data manipulation. The environment is set up to mimic a two-dimensional space with various obstacles.

2.12. Motion Model Description

The motion model is formulated by the possible movements the robot can take through the grid. It is defined by a set of tuples representing the movement cost and energy consumption associated with each movement direction [x, y] as in Table 1. The values in table 1 were selected experimentally to reflect the MCDM problem.

Table 1 - Motion model

DOI: https://doi.org/10.60797/COMP.2024.3.2.2

Direction [x, y]	Energy Consumption	Movement Cost
Right [1, 0]	1	1
Left [-1, 0]	1.2	1
Up [0, 1]	1.1	1
Down [0, -1]	1.3	1
Top-right [1, 1]	1.9	0.5
Top-left [-1, 1]	1.7	0.5
Bottom-right [1, -1]	1.8	0.5
Bottom-left [-1, -1]	1.6	0.5

2.13. MaGDR Criteria and weights

The set of options (Nodes in search space) and Criteria (Movement cost, Euclidean distance to the goal, Energy) was created for every step of the robot towards the goal. Weights were assigned experimentally as [0.1, 0.7, 0.2]; varying these values could yield different results.

Results and Analysis

In evaluating the performance of the MaGDR and Dijkstra algorithms, the metrics used are as follows:

1. Path Length: This metric measures the total nodes covered by the path selected by each algorithm from the start to the goal node.

2. Energy Consumption: Reflecting the cumulative energy cost associated with the entire path.

3. Movement Cost: This is the cumulative cost along the selected path.

4. Number of Visited Nodes (Search Space): This counts the total number of nodes each algorithm evaluates before finalizing the final path. It is an indicator of the algorithm's Decision-Making efficiency.
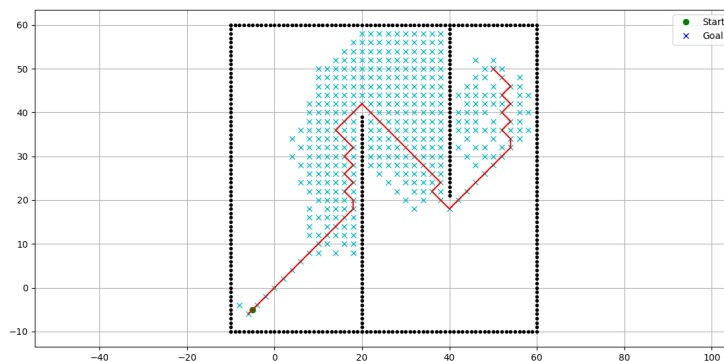

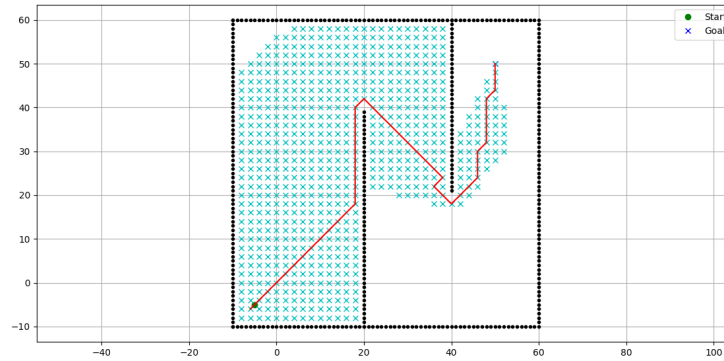Figure 2 - MaGDR selected path
DOI: https://doi.org/10.60797/COMP.2024.3.2.3

Figure 3 - Dijkstra selected path
DOI: https://doi.org/10.60797/COMP.2024.3.2.4

Comparative Analysis

Table 2 - Performance Metrics Comparison

DOI: https://doi.org/10.60797/COMP.2024.3.2.5

Metric	MaGDR	Dijkstra
Path Length (nodes)	52	52
Energy Consumption	93.8	79.8
Movement Cost	27	37
Number of Visited Nodes	705	1230

1. Path Length: Both algorithms identified paths comprising 52 nodes, indicating an equality in terms of the shortest path capability under the tested conditions.

2. Energy Consumption: MaGDR recorded a higher energy consumption of 93.8 compared to 79.8 by Dijkstra. Notably MaGDR prioritize the shorter paths based on the value of weight (user preferences).

3. Movement Cost: Dijkstra's path had a higher cumulative cost of 37, compared to 27 by MaGDR. This proves that MaGDR is more cost-efficient based on the selected parameters.

4. Number of Visited Nodes: Dijkstra explored a larger search space with 1230 nodes visited, significantly more than the 705 by MaGDR. This indicates that MaGDR is more decision-making efficient, evaluating fewer nodes before concluding the path planning.

Conclusion

The development and implementation of the MaGDR algorithm, incorporating the Mahalanobis distance, represents a significant advancement in the field of MCDM. This research demonstrates that the MaGDR algorithm provides a robust, scale-invariant method for evaluating complex decision-making scenarios. By addressing the inherent limitations of traditional GDR algorithms, particularly in handling correlations and scaling issues among different criteria, MaGDR offers more general uses. The application of the MaGDR algorithm in robotic pathfinding has verified its potential to improve decision-making efficiency. The simulation results clearly show that MaGDR can effectively navigate environments, optimize multiple conflicting criteria. Although MaGDR sometimes results in higher energy consumption, its ability to minimize movement costs and evaluate fewer nodes before reaching a decision underscore its efficacy in strategic pathfinding applications. Future research could further explore the scalability of MaGDR across other domains such as healthcare, environmental management, and logistics, where decision-making scenarios are complex.

Список литературы / References

1. Taherdoost H. Multi-Criteria Decision Making (MCDM) Methods and Concepts / H. Taherdoost, M. Madanchian // Encyclopedia. — 2023. — 1. — p. 77-87. DOI: 10.3390/encyclopedia3010006.

2. Aruldoss M. A Survey on Multi Criteria Decision Making Methods and Its Applications / M. Aruldoss, T.M. Lakshmi, V.P. Venkatesan // American Journal of Information Systems. — 2013. — 1. — p. 31-41.

3. Sahoo S.K. A Comprehensive Review of Multiple Criteria Decision-Making (MCDM) Methods: Advancements, Applications, and Future Directions / S.K. Sahoo, S.S. Goswami // Decision Making Advances. — 2023. — 1. — p. 25-48. DOI: 10.31181/dma1120237.

4. Перепелица В.А. Математические модели и методы оценки рисков экономических, социальных и аграрных процессов / В.А. Перепелица, Е.В. Попова — Ростов-н/Д: Изд-во Рост. ун-та, 2002. — 207 с.

5. McLachlan G. J. Mahalanobis distance / G. J. McLachlan // Resonance. — 1999. — 6. — p. 20-26.

6. Попова Е. В. Математические модели и методы оценки рисков социально-экономических процессов : дис. ...д-ра : 08.00.13 : защищена 2002-10-10 : утв. 2003-02-11 / Е. В. Попова — Черкесск: 2003.— 240 с.

7. Zhang Z. Pseudo-inverse multivariate/matrix-variate distributions / Z. Zhang // Journal of Multivariate Analysis. — 2007. — 8. — p. 1684-1692. DOI: 10.1016/j.jmva.2006.04.002.

8. Sánchez-Ibáñez J. R. Path Planning for Autonomous Mobile Robots: A Review / J. R. Sánchez-Ibáñez, C. J. Pérez-del-Pulgar, A. García-Cerezo // Sensors. — 2021. — 23. — p. P. 7898. DOI: 10.3390/s21237898.

9. Aziz A. A Comparative Analysis among Three Different Shortest Path-finding Algorithms / A. Aziz, S. Tasfia, M. Akhtaruzzaman // 2022 3rd International Conference for Emerging Technology (INCET). — 2022. — 1. — p. 1-4. DOI: 10.1109/INCET54531.2022.9824074.

10. Python. — 2024. — URL: https://www.python.org/ (accessed: 10.06.2024).

Список литературы на английском языке / References in English

1. Taherdoost H. Multi-Criteria Decision Making (MCDM) Methods and Concepts / H. Taherdoost, M. Madanchian // Encyclopedia. — 2023. — 1. — p. 77-87. DOI: 10.3390/encyclopedia3010006.

2. Aruldoss M. A Survey on Multi Criteria Decision Making Methods and Its Applications / M. Aruldoss, T.M. Lakshmi, V.P. Venkatesan // American Journal of Information Systems. — 2013. — 1. — p. 31-41.

3. Sahoo S.K. A Comprehensive Review of Multiple Criteria Decision-Making (MCDM) Methods: Advancements, Applications, and Future Directions / S.K. Sahoo, S.S. Goswami // Decision Making Advances. — 2023. — 1. — p. 25-48. DOI: 10.31181/dma1120237.

4. Perepelitsa V.A. Matematicheskie modeli i metody otsenki riskov ekonomicheskih, sotsial'nyh i agrarnyh protsessov [Mathematical Models and Methods for Assessing the Risks of Economic, Social and Agrarian Processes] / V.A. Perepelitsa, E.V. Popova — Rostov-n/D: Izd-vo Rost. un-ta, 2002. — 207 p. [in Russian]

5. McLachlan G. J. Mahalanobis distance / G. J. McLachlan // Resonance. — 1999. — 6. — p. 20-26.

6. Popova E. V. Matematicheskie modeli i metody otsenki riskov sotsial'no-ekonomicheskih protsessov [Mathematical models and methods for assessing the risks of socio-economic processes] : dis....of PhD in Natural sciences : 08.00.13 : defense of the thesis 2002-10-10 : approved 2003-02-11 / E. V. Popova — Cherkessk: 2003.— 240 p. [in Russian]

7. Zhang Z. Pseudo-inverse multivariate/matrix-variate distributions / Z. Zhang // Journal of Multivariate Analysis. — 2007. — 8. — p. 1684-1692. DOI: 10.1016/j.jmva.2006.04.002.

8. Sánchez-Ibáñez J. R. Path Planning for Autonomous Mobile Robots: A Review / J. R. Sánchez-Ibáñez, C. J. Pérez-del-Pulgar, A. García-Cerezo // Sensors. — 2021. — 23. — p. P. 7898. DOI: 10.3390/s21237898.

9. Aziz A. A Comparative Analysis among Three Different Shortest Path-finding Algorithms / A. Aziz, S. Tasfia, M. Akhtaruzzaman // 2022 3rd International Conference for Emerging Technology (INCET). — 2022. — 1. — p. 1-4. DOI: 10.1109/INCET54531.2022.9824074.

10. Python. — 2024. — URL: https://www.python.org/ (accessed: 10.06.2024).