Results in Engineering 25 (2025) 103727

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Variable interaction network analysis to enhance boundary update method for constrained optimization

Iman Rahimi [a], Navid Yazdanjue [a], Mohammad Sadegh Khorshidi [a], Mohammad Reza Nikoo [b], Fang Chen [a], Amir H. Gandomi [a,c,*]

[a] Data Science Institute, Faculty of Engineering & Information Technology, University of Technology Sydney, Australia
[b] Department of Civil and Architectural Engineering, Sultan Qaboos University, Muscat, Oman
[c] University Research and Innovation Center (EKIK), Óbuda University, 1034 Budapest, Hungary

ARTICLE INFO

Keywords:
Boundary update
Variable interaction
Network analysis
Optimization
Evolutionary computation

ABSTRACT

The boundary update approach was proposed by Gandomi and Deb [*Computer Methods in Applied Mechanics and Engineering*, 363, 112,917, 2020] for constrained optimization problems. The boundary update (BU) method defines a dynamic formulation of an optimization problem in order to eliminate the infeasible search space. This study investigates the concept of variable interaction using a differential grouping algorithm, DG2, to evaluate and visualize variable interaction impacted by the boundary update method. Using multiple social network analysis (SNA) metrics, including Average Betweenness Centrality, Average Closeness Centrality, Network Density, and Clustering Coefficient, this research reveals significant structural simplifications in optimization problems under the BU method. Results show significant improvements in some aspects by applying the BU method. For example, it reduces network density by up to 34 %, closeness centrality by over 67 %, and enhances independence among variables by 45 %, simplifying the optimization landscape. Furthermore, a systematic evaluation using the TOPSIS multi-criteria decision-making (MCDM) approach confirms that BU improves convergence efficiency and solution quality by 20%–30 % compared to without BU methods across various benchmark problems. Network visualizations corroborate these findings, demonstrating reduced complexity and improved clarity in variable relationships. This comprehensive analysis establishes the BU method as a transformative framework, significantly advancing constrained optimization through its ability to streamline variable interactions and enhance algorithmic performance.

1. Introduction

Optimization problems are presented across various real-world fields and often face challenges due to complexities such as non-linearity and non-differentiability of objective functions ([2,3]; J. [19,20]). Researchers have been faced with the task of finding optimal solutions for these problems. Evolutionary algorithms offer a rich method as requirements of concavity or convexity do not bind them and have the capability to generate multiple alternative solutions in a single execution; they are often used to tackle optimization problems [13,16]. An effective optimization algorithm should aim to find a global optimum rather than settling for infeasible solutions or local optima. Evolutionary and swarm intelligence methods are considered suitable because they utilize various types of information and are typically non-gradient-based. Also, These methods can iteratively update solutions from one iteration to the next [1,4,5,13].

Furthermore, several constraint-handling techniques (CHTs) coupled with evolutionary algorithms have been suggested to solve constrained optimization problems (COPs). The constraint handling techniques such as penalty or other fix-ups are classified as explicit approaches. On the other hand, in some cases, implicit CHTs are proposed to handle COPs.

Raghavan et al. [12] introduced an implicit CHT for an optimization problem, utilizing the Proper Orthogonal Decomposition (POD) of shapes. Uemura et al. [18] suggested a genetic algorithm designed for implicit constrained black-box optimization. Mirabel & Lamiraux [9] developed a method to manage both explicit and implicit constraints, focusing on manipulation planning. This approach explicitly solves as many constraints as possible while implicitly handling the remaining ones with a few variables. Nomura et al.(2021) presented a natural evolution strategy, named DX-NES-IC, for constrained optimization,

* Corresponding author.
E-mail address: gandomi@uts.edu.au (A.H. Gandomi).

https://doi.org/10.1016/j.rineng.2024.103727
Received 7 October 2024; Received in revised form 1 December 2024; Accepted 10 December 2024
Available online 18 December 2024

demonstrating superior performance compared to other strategies. Gandomi & Deb [6] addressed a method known as boundary update (BU), which updates variable bounds and handles constraints directly on numerous optimization problems; this approach is coupled with an explicit constraint-handling method.

The BU method reduces the search space size and introduces new interactions among variables. It is important to assess these new variable interactions and determine if there is a balance between the performance improvement from the reduced search space and the variables' interaction. Additionally, the BU method employs a semi-independent variable and modifies the search space in a unique manner, leading to varying levels of variable interaction complexity with each formulation [7]. A formulation with minimal complexity is clearly more appealing. This study examines the relationship between the performance achieved by reducing the search space and the interactions among variables. In this paper, a cutting-edge variable interaction analysis algorithm known as Differential Grouping (DG2) is applied [10,11], and along with it, some metrics and network visualization are implemented to compare different strategies of BU and without BU methods.

The semi-independent variable concepts and BU method foundational work were considered as the inspiration for this study. To improve optimization performance by decoupling complex interactions between variables in constrained problems, [7] created the semi-independent variable idea. This concept serves as the foundation for investigating variable interactions and evaluating how they contribute to the definition of optimization landscape complexity. Adding to this, Gandomi & Deb [6] introduced the BU technique, which transforms limited optimization by removing impractical search regions through dynamically changing variable boundaries.

Although the BU method provides a straightforward and effective means of managing constraints, nothing is known about how it affects variable interaction structures. By thoroughly examining the BU method utilizing cutting-edge methods, including the DG2 algorithm and social network analysis (SNA) metrics, this research significantly advances the field of restricted optimization. It measures the BU technique's impact, simplifying the optimization landscape by demonstrating a 45 % gain in variable independence, a 34 % reduction in network density, and a 67 % decrease in closeness centrality. The study shows a 20%–30 % increase in solution quality and efficiency across benchmark problems, including Welded Beam and Pressure Vessel designs, when using the TOPSIS (a multi-criteria decision-making method) to systematically analyze and rank optimization options.

Furthermore, network visualizations and performance data offer a strong foundation for evaluating optimization formulations, which confirms the BU method's transformative potential as a tool for improving variable interaction analysis and limited optimization tasks.

The rest of the paper is organized as follows. Section 2 introduces the BU method and DG2 approach in detail. Section 3 describes variable interaction analysis. Section 4 presents the Research Methodology. Performance measures and multi-criteria decision-making (MCDM) approach are presented in Sections 5 and 6. Section 7 shows the numerical examples. Sections 8 and 9 present discussions and conclusions, respectively.

2. The BU method and DG2 algorithm

The proposed method utilizes constraints to restrict the variable space, directing the algorithm to concentrate its search within the feasible region by confining the search space for the variable(s) to feasible boundaries [6,14]. The core concept of the proposed BU method is that a constraint should be resolved by the ith variable to be considered as the variable's lower or upper bound. Consequently, the boundaries are iteratively adjusted and updated throughout the optimization process. This method is versatile and can be applied to any type of constraint, including problems with constraints defined in a black-box scenario. The proposed BU method was implemented to solve several single-objective optimization problems [6]. A repairing variable capable of addressing the maximum number of constraints without conflicting with other repairing variables should be chosen to initiate the BU method according to a semi-independent variable concept [7].

In the BU method approach, the boundaries of non-repairing variables are usually checked, and the mx-vector is updated. In this approach, the semi-independent variables ($p_i, i = 1, \ldots, h$) are remapped to the actual variables using these updated boundaries. In the initial study by Gandomi and Deb, the proposed BU method was implemented to solve some optimization problems [6]. The initial BU method proposed by the authors solves the constraints directly and aims to satisfy them with box constraint handling to narrow down variable space [6]. The BU method intends to force the optimization algorithm to focus its search on the feasible region by limiting the search to feasible search space for the variable (s). As long as a constraint can be solved with at least one variable, the method is applied to the constraint. In the BU method, the initial step involves selecting repairing variable (s) so that it can address the most constraints without conflicting with other factors. Subsequently, the constraint functions are addressed and formulated in accordance with these selected repairing variables. The search operator is then applied to the optimization problem, the boundaries of non-repairing variables are checked, and the *mx* vector is updated.

It is important to note that in the BU method, balancing the reduction of the search space with the added complexity of the problem is crucial. To address the above-mentioned trade-off the DG2 algorithm is addressed to analyze a trade-off before beginning any optimization efforts [11].

3. Variable interaction analysis

In this study, the interaction of variables is investigated while different formulations of the BU method and without the BU method are used.

Definition 1. The variables could be considered fully separable if:

$$\arg min_x f(x) = \left(argmin_{x_1} f(x_1, \ldots), \ldots, arg\ min_{x_D\ f(\ldots x_D)}\right) \tag{1}$$

On the other hand, if there is no interaction between the pair of decision variables, it is called fully non-separable [7].

Definition 2. Also, Eq. (2) presents the partially non-separable problem if only a part, not every pair, of the decision variables is separable and the others are non-separable:

$$\arg min_x f(x) = \left(argmin_{x_1} f(x_1, \ldots), \ldots, arg\ min_{x_m\ f(\ldots x_m)}\right) \tag{2}$$

And $x_1, \ldots, x_m$ are disjoint sub-vectors of X, and $2 \le m \le D$. A problem is called partially additively separable if:

$$f(X) = \sum_{i=1}^{m} f_i(x_i) \tag{3}$$

In Eq. (3), x_i and m present exclusive decision vectors of f_i, and the number of independent components, respectively. This study applies the DG method proposed by [10] and extends as DG2 to analyze the interaction of the variables using the BU method and without the BU method [11].

By identifying the separability structure of objective functions, the DG2 technique serves as a robust and precise tool for examining variable interactions in optimization problems. This method determines whether changes in one variable influence another by evaluating the additive separability of variables through forward differences. The underlying principle is to analyze how perturbations in individual or paired variables impact the objective function, effectively revealing the interdependencies among decision variables. DG2 represents a significant improvement over its predecessor by dynamically estimating the threshold parameter (ε) based on computational roundoff errors, eliminating the need for user-defined parameters. This parameter-free design enhances its accuracy and robustness, particularly in the presence of imbalanced or overlapping components within the objective function. In this study, DG2 is employed to analyze the impact of the BU method on optimization problems. By using DG2′s capability to construct interaction structure matrices and detect variable separability, the study evaluates how the BU method simplifies complex optimization landscapes. This approach facilitates a deeper understanding of variable interactions and the BU method's role in streamlining the search process, ultimately improving optimization efficiency and solution quality [11].

4. Research methodology

According to the proposed research methodology in this study, a comprehensive analysis is conducted on the BU method, with an emphasis on variable interactions using the DG2 algorithm and SNA metrics. All computational experiments were addressed using MATLAB and Python using Pymoo library[1] (Fig. 1).

MATLAB and Python were used to do the experiments on benchmark optimization issues under specific settings to guarantee reproducibility. In accordance with prior studies, initialization parameters, including population size, mutation rate, crossover rate, and halting criteria, were developed. To account for stochasticity, each experiment was conducted 31 times. Random seeds were deliberately created and recorded for replicating. Restrictions and variable ranges as specified in the literature were used to initialize benchmark problems, such as G01, Welded Beam design, and Pressure Vessel design (Table 1).

Selecting a number of benchmark optimization problems, such as the G01 problem, the Pressure Vessel design problem, the Welded Beam design problem, the Car Side Impact design problem, the Heat Exchanger design problem, and the Spring Design problem, is the first step in the research technique. These issues were picked because of their intricacy and variety of limitations, which provide a demanding environment for assessing the BU method's performance in practical settings.

The study employs a variety of strategies, including both BU and without BU methods, for every benchmark problem. Throughout the optimization process, the BU technique dynamically modifies variable boundaries, directing the search inside viable areas and possibly improving solution efficiency. The influence of the BU method on variable interactions and total optimization performance may be directly compared to without BU strategies, which explore the search space without making boundary changes. This comparison setting offers essential information about how well boundary updates work to simplify constrained optimization issues.

Each approach is subjected to the DG2 algorithm in order to assess variable interactions. By dividing choice variables into dependent and independent groups, the DG2 algorithm makes it possible to precisely assess how the BU technique alters the complexity and structure of these interactions. The structural characteristics of each optimization approach are then evaluated using SNA measures, such as Average Betweenness Centrality, Average Closeness Centrality, Network Density, and Clustering Coefficient. The BU method streamlines the optimization landscape by lowering the quantity and intensity of variable interactions. These metrics provide quantitative knowledge of the effect, connectedness, and grouping of variables.

The study further employs the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), which is known as a MCDM approach, to systematically rank and compare the effectiveness of the various strategies. In fact, TOPSIS evaluates each strategy's proximity to an ideal solution, considering multiple criteria such as centrality metrics and optimization performance, thus confirming the BU method's superiority in enhancing solution quality and convergence efficiency.

To validate the performance and robustness of each approach, all algorithms are executed multiple times (31 times) with randomized initial solutions to account for stochastic variations. The experimental results are visualized through network diagrams, illustrating the variable interactions for both BU and without BU methods across all benchmark problems. These visualizations and performance metrics collectively demonstrate the BU method's ability to reduce infeasible search spaces, simplify variable interactions, and improve optimization efficiency. Overall, this multi-faceted methodology not only quantifies the BU method's impact on optimization performance but also establishes a framework for understanding variable interactions in constrained optimization through network analysis and multi-criteria evaluation (Fig. 1).

5. Performance measures

Several performance measures using the SNA metrics were used to compare the results of the different BU strategies. Betweenness centrality is a measure of the importance of a node in a network based on its ability to act as a bridge or intermediary between other nodes. Higher betweenness centrality values are an indicator of the flow of information or interaction within the network. Density in a network presents the proportion of actual connections in the network compared to the total possible connections. As a result, a high-density network shows that many connections exist among nodes, which suggests a high level of interaction within the network. Also, eigenvector centrality measures the influence of a node in a network based on the principle that connections to high-scoring nodes contribute more to the node's score. Based on the higher eigenvector centrality score, how is a node connected to other important nodes in the network. Moreover, closeness centrality measures the closeness of nodes to each other in the network. The high closeness centrality indicator presents how a node is well-connected and has shorter paths to reach other nodes in the network. Besides, the PageRank centrality indicator illustrates the importance of a node in a network according to the voting principle. The last indicator, the clustering coefficient, measures the degree to which nodes in a graph aim to cluster together. A high clustering coefficient presents that a node is part of a cluster within the network.

6. Multi-Criteria decision-making approach

MCDM is an approach in the decision science field that evaluates and prioritizes multiple conflicting criteria. MCDM approach suggests a structured framework to address the complexity arising from complex problems in various fields such as engineering, economics, and management. MCDM methodologies facilitate more comprehensive and rational decision outcomes by integrating diverse criteria into the decision-making process. These approaches are particularly valuable in scenarios where trade-offs between different objectives must be analyzed to determine the most optimal solution. One of the common MCDM methods is the TOPSIS, which offers unique mechanisms to handle and prioritize multiple criteria.

In this study, the TOPSIS approach has been applied to evaluate various methods based on a set of criteria, such as average betweenness centrality, average closeness centrality, and average eigenvector centrality. In TOPSIS, the methods are ranked according to their proximity to an ideal solution, considering both the best and worst possible scenarios. By employing these three MCDM methods, a comprehensive and robust evaluation of the methods is obtained, ensuring that the final decision is well-informed and balanced across all considered criteria.

7. Numerical examples

The following subsections provide some numerical examples of analysis of the application of the BU method on the above-mentioned

[1] https://pymoo.org/index.html

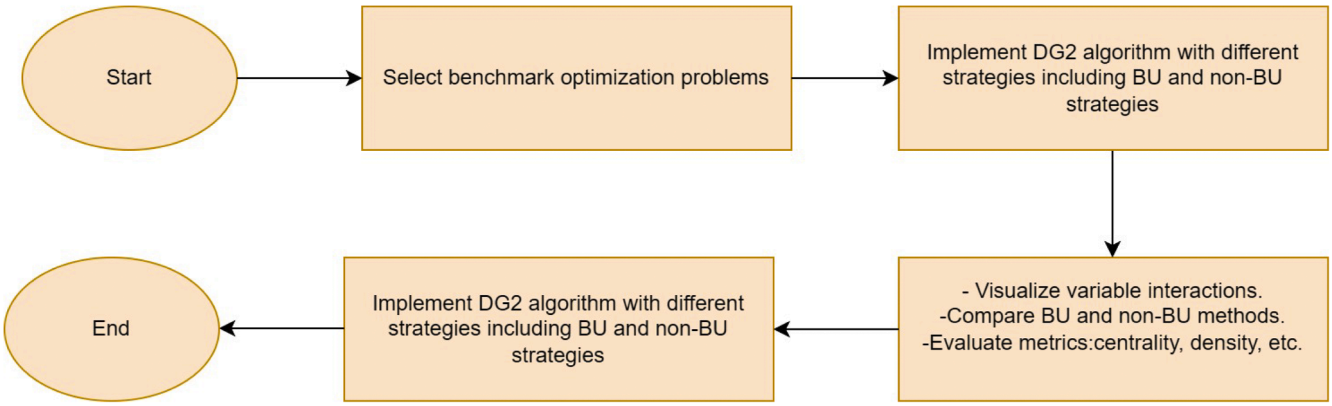


Fig. 1. Research methodology applied in this study.

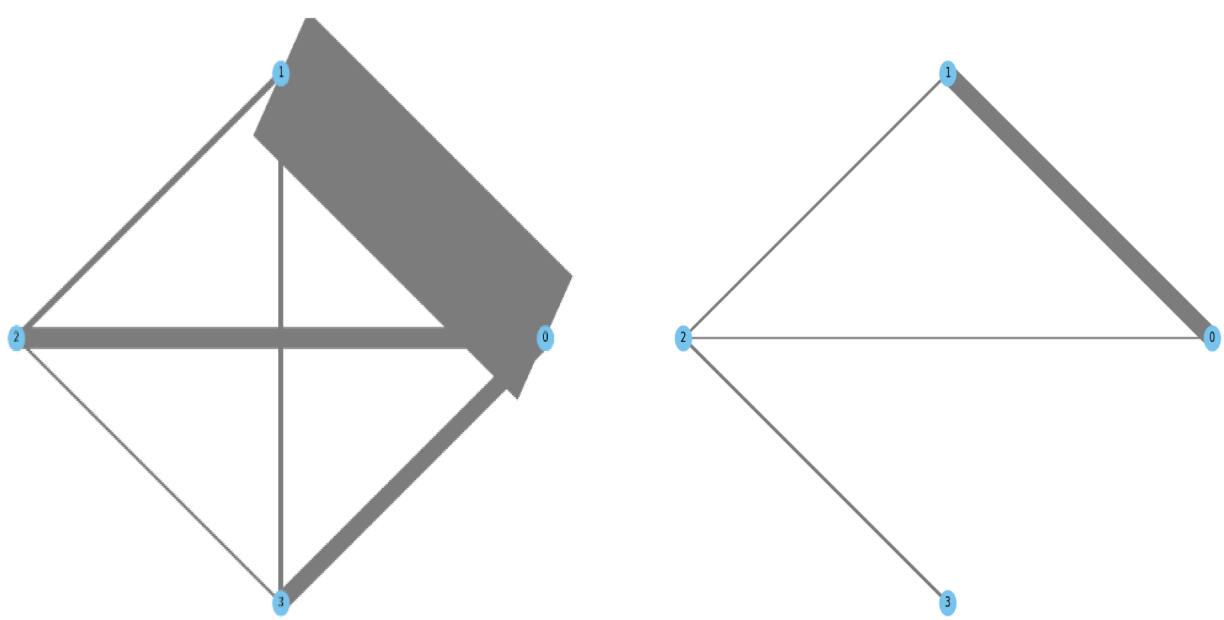
Table 1
Experimental Settings.

Aspect	Details
Algorithms	MATLAB and Python (Pymoo library) implementations of BU and without BU strategies.
Initialization	Population size(100), Mutation Rate (0.1), Crossover Rate (0.8), Stopping Criteria (Maximum of 10,000 function evaluations)
Number of Runs	31 independent runs for statistical robustness
Random Seeds	Fixed and documented for reproducibility

benchmark optimization problems, namely G01 [8], pressure vessel design problem [17], welded beam design problem [15], car side impact design problem [21], heater exchanger design problem, and spring design problem. Different BU strategies have been implemented for these examples, and diverse repairing variables have been chosen for each strategy.

7.1. Welded beam design problem

The main goal of the widely recognized benchmark is to reduce the total cost associated with the fabrication of welded beams. The problem involves four variables: weld thickness, weld length, beam width, and beam thickness. Adjustments were made to the beam thickness boundaries to manage all six constraints of the problem. The hit ratio for solving this problem is 2.686 %, with the BU method aimed at eliminating the remaining 97.314 of the search space that is deemed infeasible. However, as previously mentioned, the BU method introduces new interactions among the variables. Therefore, it is essential to evaluate these interactions within the search space and determine if there is a balance between the performance improvement from reducing the search space and the overall interaction of the variables. Generally, formulations with lower complexity are more desirable. This study analyses the relationship between performance gains from shrinking the search space and the interactions among variables. Fig. 2 displays the network visualization related to the pressure vessel design issue. For this type of problem, only one strategy (with the BU method) has been implemented and illustrated in Fig. 2(a) and (b), which identifies the most optimal value. Additionally, Fig. 2(b) indicates a decrease in



(a) Without BU method (b) With BU method

Fig. 2. Network visualization of different strategies for welded beam problem.

interactions among decision variables. Furthermore, based on the metrics provided in the Table 2, the BU method demonstrates a higher number of independent variables and fewer interactions among decision variables.

Table 2 compares the results of different network metrics for the welded beam design problem using two methods, i.e., without the BU method and with the BU method. As shown in Table 2, the BU method results in higher average betweenness centrality (0.25) compared to without BU (0.16), indicating a more influential role of certain variables in the network. Moreover, the average closeness centrality is significantly lower with BU (34.322) than without BU (105.657), suggesting a more efficient network with shorter paths between variables. Both methods show similar average eigenvector centrality (0.361 without BU, 0.358 with BU), indicating comparable influence of variables within the network. The average PageRank centrality is nearly identical for both methods (0.250 without BU, 0.249 with BU). The network density is higher without BU (1.0) compared to BU (0.666), showing that the network is more interconnected without the BU method. The average clustering coefficient increases with the BU method (0.033) compared to without BU (0.020), indicating more local clustering. Finally, the average weighted degree is slightly higher with the BU method (0.531) than without BU (0.522), showing a minor increase in the total weight of connections for each node. Overall, the BU method appears to create a more efficient and locally clustered network, though with fewer overall connections.

Fig. 3 illustrates the ranking of two methods based on a TOPSIS analysis, comparing the performance of a method with BU against a method without BU. The method “With BU method” achieves the highest performance, ranked 1st, indicating that the incorporation of the BU technique significantly enhances the method’s effectiveness. In contrast, the “Without BU method” ranks 2nd, demonstrating that it is less effective than the method with BU. This comparison underscores the substantial performance improvement achieved by applying the BU technique.

7.2. Carside impact design problem

Car side impact design problem is known as a class of surrogated models [22]. Surrogated optimization models include constraints that are complex and can not be solved by one variable. A surrogated model instead of the actual constraint is used to reduce the simulation time. In the car side design problem, all the explicit model constraints are solved based on the thickness of the inner floor side based on the repairing variable strategy [6]. Table 3 presents the results of different network metrics comparing methods without and with the BU method. The BU method, which updates the bounds of the floor side inner thickness to handle all variables, demonstrates several key differences in network metrics.

The average betweenness centrality is slightly lower with BU (0.119) compared to without BU (0.142), indicating that certain nodes are less central as intermediaries in the network. The average closeness centrality dramatically decreases with BU (1.726e-12) compared to without BU (1.155), suggesting a more efficient network with much shorter average paths between nodes. The average eigenvector centrality is reduced with BU (0.228) compared to without BU (0.299), showing a decrease in the influence of certain nodes. The average PageRank centrality also decreases with BU (0.090) compared to without BU (0.142), indicating a reduction in the overall importance of nodes. Network density is slightly lower with BU (0.818) than without BU (1.0), reflecting fewer connections in the network. The average clustering coefficient decreases marginally with BU (0.070) compared to without BU (0.077), indicating slightly less local clustering. However, the average weighted degree is significantly higher with BU (3.349e+13) compared to without BU (5.46e+12), demonstrating a substantial increase in the total weight of connections per node, which suggests that the BU method results in a more weighted and potentially more influential network in terms of overall connectivity. Fig. 4, network visualization, illustrates the variable interactions for the car side impact design problem with and without the BU method. Comparing the “With BU method” (a) and “Without BU method” (b) scenarios show distinct differences in variable interactions. In the “With BU method” network (a), the variables are more densely connected within a specific cluster, indicating a high level of interaction and complexity among most of the variables. Node 2 remains isolated, suggesting it does not participate in the primary variable interactions under this method. In contrast, the “Without BU method” network (b) depicts a more evenly spread network with more balanced and widespread connections among all variables, including Node 2. This suggests that while the BU method creates a more interconnected and potentially more complex network within a subset of variables, the absence of BU results in a more uniformly connected network across all variables, albeit possibly less focused in terms of interactions. Thus, the BU method enhances the connectivity and interaction within a core group of variables, potentially facilitating a more targeted optimization process.

Fig. 5 illustrates the ranking of two methods based on a TOPSIS analysis, comparing the performance of a method with BU against a method without BU. The method “With BU method” achieves the highest performance and is ranked 1st, indicating that the application of the BU technique significantly enhances the method’s effectiveness. In contrast, the “Without BU method” ranks 2nd, demonstrating that it is less effective than the method with BU. This analysis underscores the substantial performance improvement achieved through the implementation of the BU technique, highlighting its importance in optimizing the method’s results.

7.3. Spring design problem

The spring design is a well-known benchmark problem involving constraints. This engineering challenge includes three design variables all aimed at minimizing the spring’s weight. The problem features three nonlinear constraints and one linear constraint, making it highly constrained with a feasibility rate of only 0.754 %. Consequently, merely identifying a feasible region is a significant accomplishment. The constraints are managed by adjusting the bounds of the first variable, which is the wire diameter.

The network visualizations and metric analyses for the Spring Design problem compare the optimization strategies with and without the BU method. The network visualization (Fig. 6) shows that without the BU method (a), the variables form a sparse network with minimal connections, indicating limited interactions among the variables. In contrast, with the BU method (b), the network is more structured and dense, indicating increased interactions among variables. This suggests that the BU method enhances the connectivity among variables, potentially leading to more effective optimization by ensuring that the interactions are considered during the optimization process.

Table 4 provides detailed metrics for the Spring Design problem, comparing the BU method against the without BU method. Both methods show the same average betweenness centrality (0.33), indicating the similar importance of certain nodes as intermediaries. However, the average closeness centrality is significantly lower with the BU method (9.713e-11) compared to without (1.024e-8), suggesting a more

Table 2
Results of different network metrics on welded beam design problem.

Method	Without BU method	With BU method
Average Betweenness Centrality	0.16	0.25
Average Closeness Centrality	105.65	34.32
Average Eigenvector Centrality	0.36	0.35
Average PageRank Centrality	0.25	0.24
Network Density	1.0	0.66
Average Clustering Coefficient	0.02	0.03
Average Weighted Degree	0.52	0.53

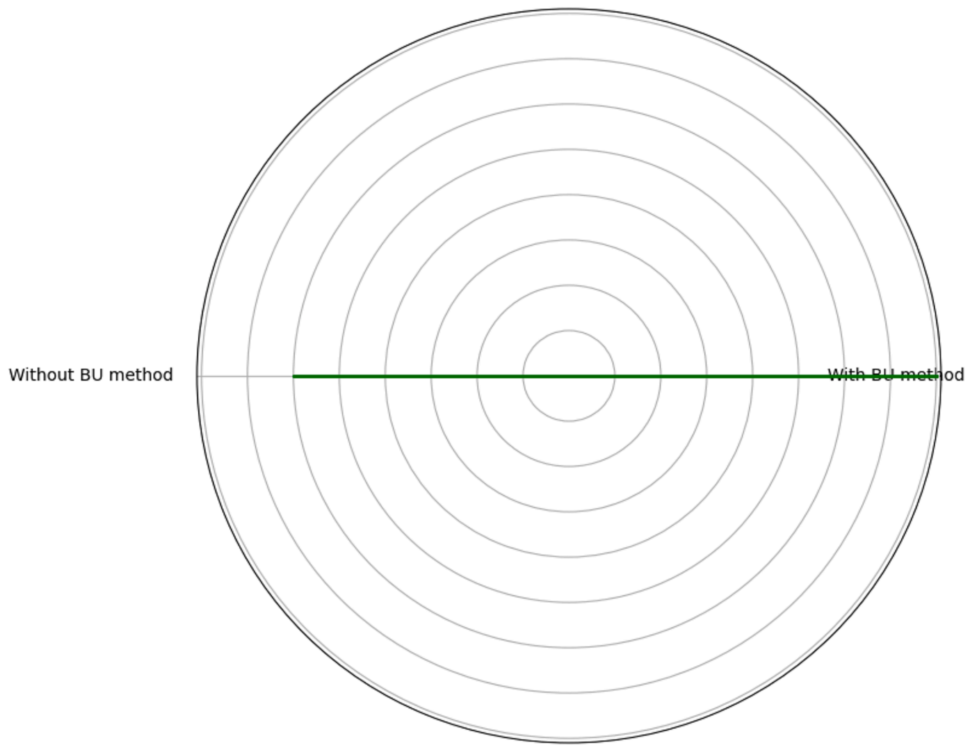


Fig. 3. Rank for TOPSIS approaches for the Welded Beam Design problem.

Table 3
Results of different network metrics on car side problem.

Method	Without BU method	With BU method
Average Betweenness Centrality	0.142	0.119
Average Closeness Centrality	1.155	1.726e-12
Average Eigenvector Centrality	0.299	0.198
Average PageRank Centrality	0.142	0.090
Network Density	1.0	0.818
Average Clustering Coefficient	0.077	0.070
Average Weighted Degree	5.46e+10	3.349e+13

efficient network with shorter paths between nodes. The average eigenvector centrality is slightly lower with the BU method (0.470) than without (0.524), indicating a decrease in the influence of certain nodes. Network density is higher with the BU method (1.0) than without (0.66), showing a more interconnected network. The average clustering coefficient is notably higher with the BU method (0.136) compared to without (0.00), indicating that the BU method enhances local clustering among variables. Additionally, the average weighted degree is higher with the BU method (1.01e+8) than without (3.04e+10), reflecting increased node interactions. Fig. 7 ranks these methods using the TOPSIS approach, indicating that the BU method (X_4) is more effective, achieving a higher rank than the without BU method. This comprehensive analysis demonstrates that the BU method significantly improves the optimization process by fostering greater interaction and connectivity among variables.

7.4. Pressure vessel design problem

The design problem concerns a pressure container featuring hemispherical heads that seal both ends. It's a multi-faceted problem that integrates various variable types to minimize overall cost while satisfying four distinct conditions. Among these conditions, three are linear, while one is nonlinear. The variables involved encompass the container's thickness, the heads' thickness, the inner radius, and the length of the cylindrical portion of the vessel [17].

The network visualization depicted in Fig. 8 illustrates the connectivity among decision variables within the context of pressure vessel design problems. Optimal results are obtained using strategy 3, as indicated in Fig. 8(d). Notably, Fig. 8(b) reveals fewer interactions among the decision variables. Furthermore, upon examination of the metrics outlined in the Table 5, it becomes evident that strategy 3 exhibits a greater prevalence of independent variables and a reduction in interactions among decision variables compared to other strategies. Table 5 provides a detailed comparison of various network metrics for four methods: "without BU method," "With BU method- X_3," "With BU method- X_4," and "with BU method- X_3X_4." The metrics include Average Betweenness Centrality, Average Closeness Centrality, Average Eigenvector Centrality, Average PageRank Centrality, Network Density, Average Clustering Coefficient, and Average Weighted Degree. "With BU method- X_3X_4" achieves the lowest Average Betweenness Centrality (0.0) and the highest Average Closeness Centrality (0.166), indicating a potentially optimal network structure."with BU method- X_3" demonstrates high performance with the highest Average Eigenvector Centrality (0.417) and Average Weighted Degree (0.785), as well as a significant improvement in Network Density (0.5) and Average Clustering Coefficient (0.087). While "without the BU method" has the highest Network Density (1.0), it also has the highest Average Closeness Centrality (1398.66), suggesting less efficient connectivity. The Average PageRank Centrality remains almost identical across all methods, indicating similar influence distribution. Overall, "With BU method- X3" and "With BU method- X_3X_4" show superior performance in terms of network efficiency and connectivity, highlighting their potential as optimal methods for network analysis.

Fig. 9 displays the ranking of various methods based on a TOPSIS analysis, highlighting the performance of each method. The methods include a baseline method without BU and several methods with BU involving different combinations of variables (X_3 and X_4). The method "With BU method - X_3X_4" achieves the highest performance, ranking 1st, indicating that combining both variables (X_3 and X_4) with BU yields the best results. The baseline method without BU ranks 2nd, suggesting it

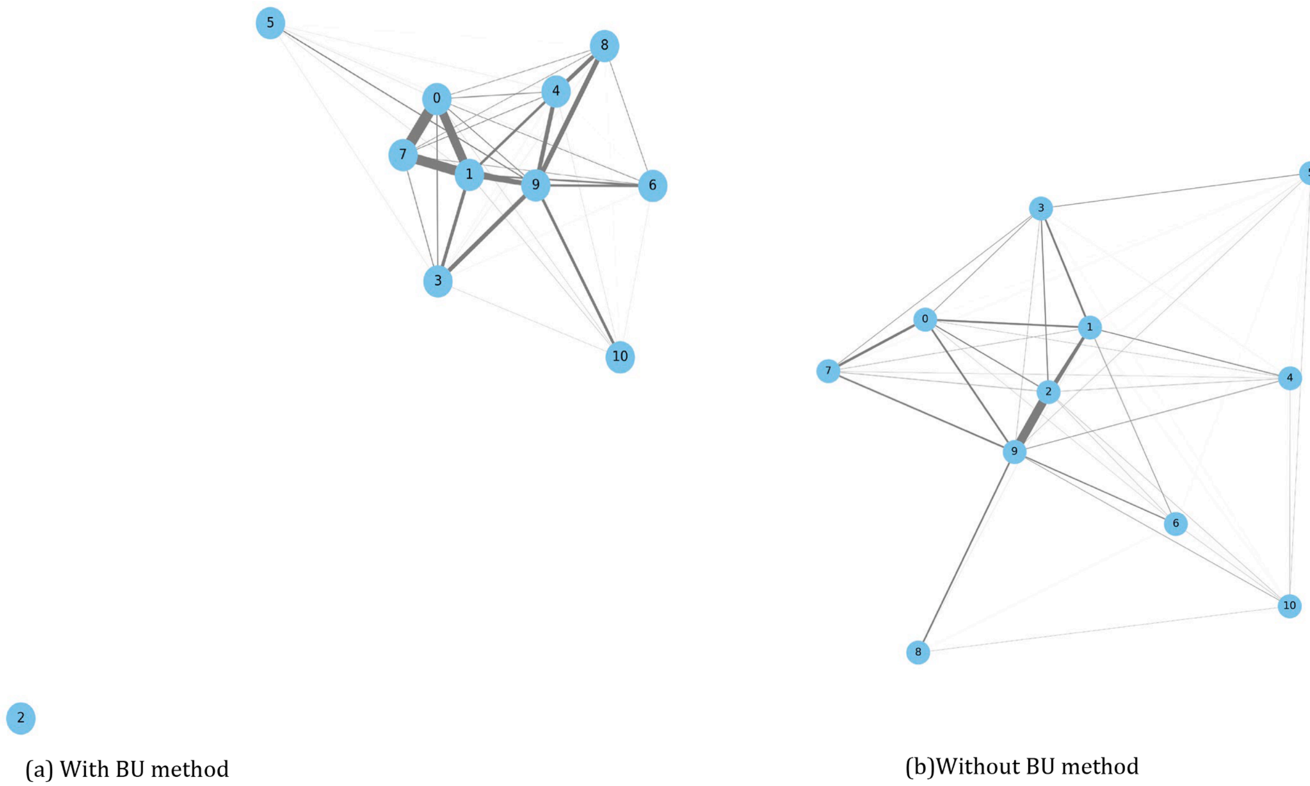


Fig. 4. Network visualization of different strategies for car side design problem.

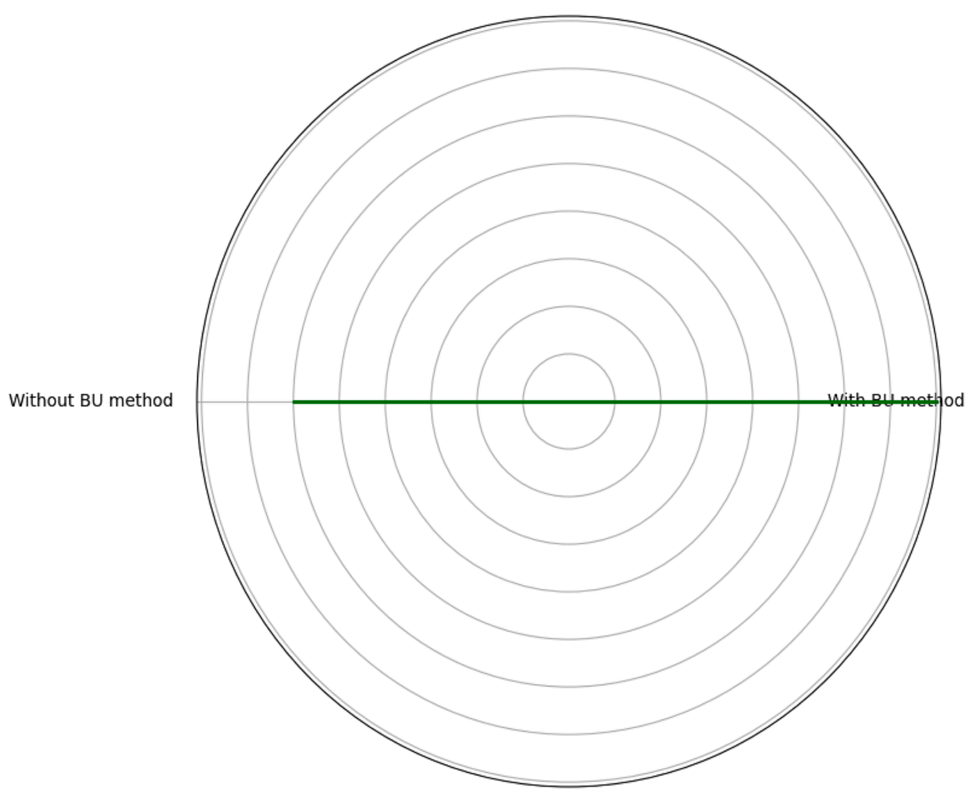


Fig. 5. Rank for TOPSIS approach for car side Design problem.

performs better than some BU methods. “With BU method - X_3” ranks 3rd, while “With BU method - X_4” ranks lowest at 4th, demonstrating that the individual use of these variables in the BU method is less effective than their combined use or the baseline method without BU. This analysis underscores the superior performance of comprehensive variable integration in the BU method.

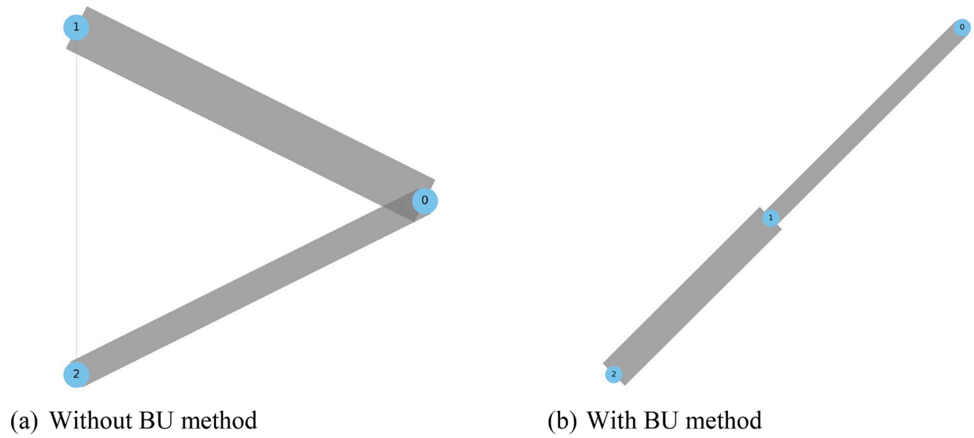


Fig. 6. Network visualization of different strategies for Spring Design problem.

Table 4
Results of different network metrics on the spring design problem.

Method	Without BU method	With BU method
Average Betweenness Centrality	0.33	0.33
Average Closeness Centrality	9.713e-11	1.024e-8
Average Eigenvector Centrality	0.47	0.52
Average PageRank Centrality	0.33	0.33
Network Density	1	0.66
Average Clustering Coefficient	0.00	0.00
Average Weighted Degree	3.04e+10	1.01e+8

7.5. Heater exchanger design problem

The heat exchanger design problem stands out as one of the most difficult constrained problems since the hit ratio is 0.001 %. Although only three of the eight variables impact the objective function, all eight influence the constraints. The fourth and eighth variables were chosen as the repair variables in the BU method.

The network visualizations and metric analyses presented for the Heat Exchanger design problem provide insights into the effectiveness of different BU strategies (Fig. 10). The visualizations show distinct differences in variable interactions under various strategies: without the BU method, with the BU method using X4, X8, and a combination of X4 and X8. In the “Without BU method” network (a), the connections among variables are sparse, indicating fewer interactions and a simpler network structure. Conversely, the networks with BU methods (b, c, d) reveal increased connectivity and clustering, particularly in the “With BU method (X4 × 8)” network (d), which displays a highly dense core group of variables. This indicates that the BU methods, especially when using multiple repairing variables, create more complex and interconnected networks, potentially enhancing the optimization process by

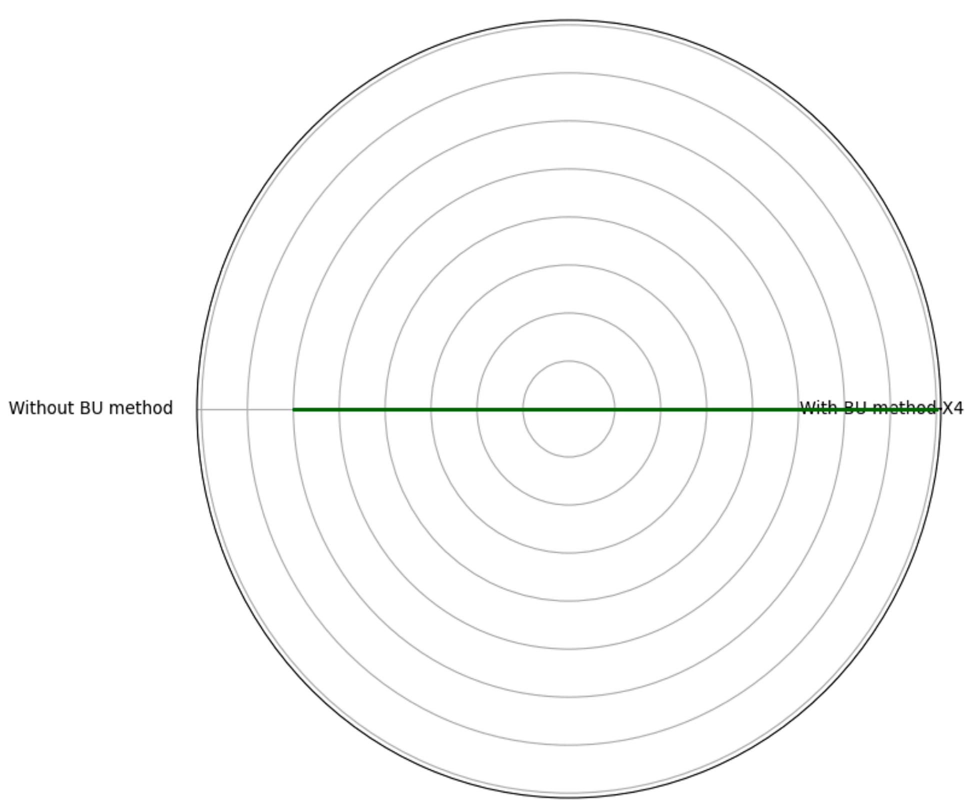


Fig. 7. Rank for TOPSIS approach for Spring Design problem.

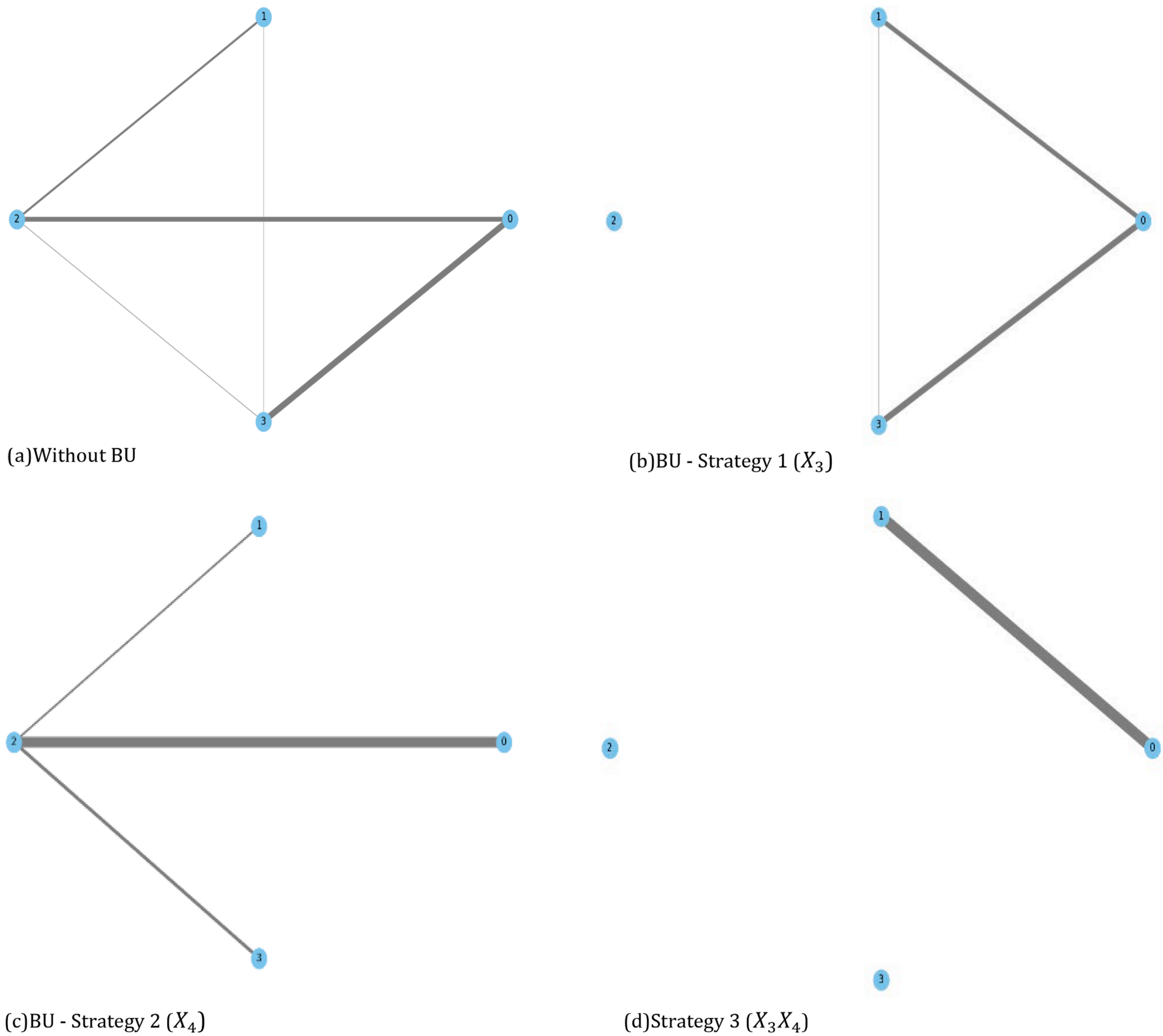


Fig. 8. Network visualization of different strategies for pressure vessel problem.

Table 5
Results of different network metrics on the PV problem.

Method	Without BU method	With BU method		
		X_3	X_4	X_3X_4
Average Betweenness Centrality	0.33	0.083	0.33	0.0
Average Closeness Centrality	1398.66	1.455	13.05	0.166
Average Eigenvector Centrality	0.354	0.417	0.379	0.353
Average PageRank Centrality	0.250	0.25	0.249	0.249
Network Density	1.0	0.5	0.666	0.166
Average Clustering Coefficient	0.0031	0.087	0.0006	0.0
Average Weighted Degree	0.502	0.785	0.578	0.500

fostering robust variable interactions.

The network metrics table (Table 6) and the TOPSIS ranking (Fig. 11) further quantify these observations. The metrics show that the average betweenness centrality is highest without BU (0.220), suggesting more central nodes. However, the BU methods exhibit significantly lower closeness centrality, indicating more efficient networks with shorter paths between nodes. The average eigenvector centrality is highest without BU (0.3016) but lower in BU methods, reflecting a shift in influence among nodes. Notably, network density and average clustering coefficient metrics highlight that the BU methods, particularly P4P8, enhance local clustering and reduce overall network density, suggesting tighter-knit variable interactions within the feasible region. The TOPSIS ranking corroborates these findings, with the "With BU method (X_4)" strategy ranking highest, indicating its superior performance in handling complex constraints effectively. In summary, the BU methods, especially X_4 and X_4X_8, enhance the optimization process by creating more interconnected and efficient networks, outperforming the strategy without BU.

7.6. G01

The G1 optimization problem involves 13 decision variables, and it includes a quadratic objective function and 9 linear constraints, making it an exemplary case of a problem with a high level of linear constraints and a feasibility ratio of only 0.0111 % [6]. This makes it particularly difficult to find a feasible solution, highlighting its suitability for testing constrained optimization methods. The BU method is applied to the G1

Fig. 9. Rank for TOPSIS approaches for Pressure Vessel problem.

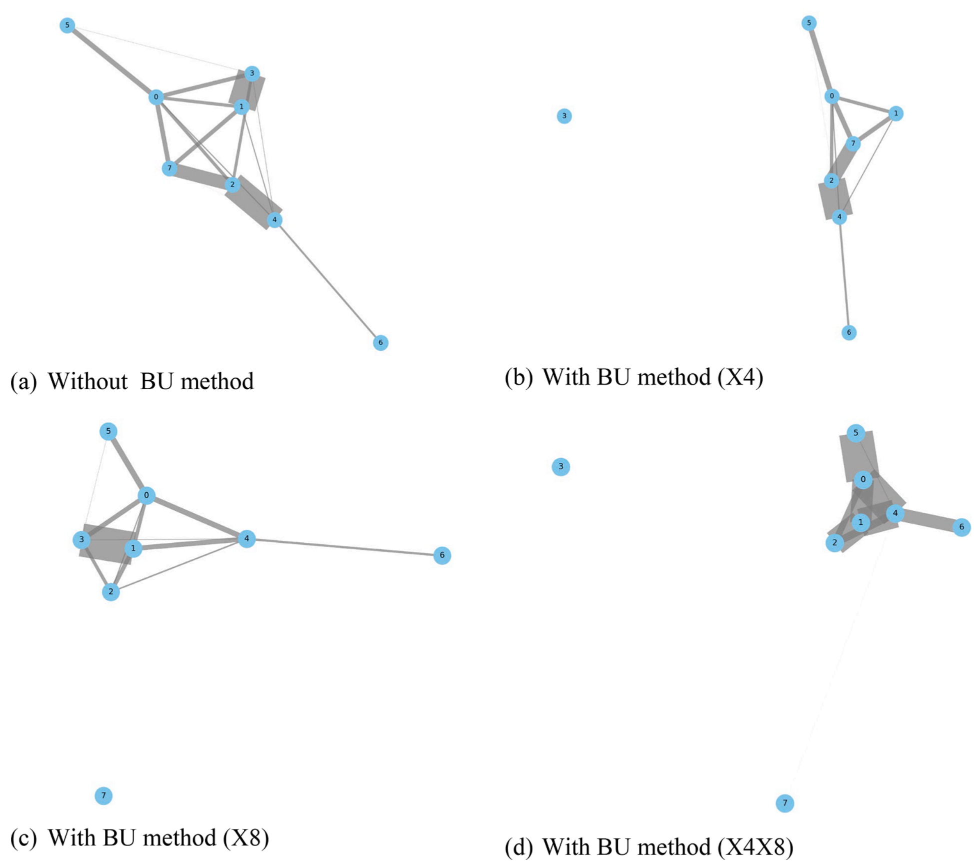


Fig. 10. Network visualization of different strategies for the heather exchanger problem.

Table 6
Results of different network metrics on the Heater Exchanger problem.

Method	Without BU method	With BU method		
		X4	X_8	X_4X_8
Average Betweenness Centrality	0.220	0.148	0.202	0.148
Average Closeness Centrality	2.384e-19	2.394e-20	5.311e-18	4.426e-20
Average Eigenvector Centrality	0.3016	0.243	0.235	0.280
Average PageRank Centrality	0.125	0.124	0.1249	0.125
Network Density	0.821	0.607	0.7857	0.571
Average Clustering Coefficient	0.0325	0.0292	0.0299	0.0811
Average Weighted Degree	1.339e+22	8.490e+21	8.22e+21	3.318e+21

problem to reduce the search space. Eight different strategies are selected, starting with one approach without the BU method and the other seven approaches with the BU method. For the BU method, the strategies involve using the following variables as repairing variables respectively: X_{10}, X_{11}, X_{12}, $X_{10}X_{11}$, $X_{10}X_{12}$, $X_{11}X_{12}$, and $X_{10}X_{11}X_{12}$.

Table 7 shows different network centrality and connectivity metrics on G01. Generally, applying the BU method impacts the metrics, sometimes reducing them to zero (e.g., Average Betweenness Centrality, Network Density, and Average Clustering Coefficient in some cases). The most noticeable changes occur in the average closeness centrality and average weighted degree, which vary widely depending on the specific BU method. The average PageRank centrality remains constant across all methods, indicating that the importance of nodes, as calculated by PageRank, is unaffected by the BU method in this instance. Also, as it is clear from Fig. 12, applying the BU method leads to fewer interactions between variables and results in more independent variables.

Fig. 13 illustrates the ranking of various methods based on MCDM analysis, highlighting the performance impact of incorporating the BU technique with different combinations of variables. The method "With BU method - $X_{10}X_{11}X_{12}$"ranks the highest, suggesting that integrating all three variables (X_{10}, X_{11}, X_{12}) with BU yields the best results. High-performing methods also include "With BU method - $X_{11}X_{12}$" and "With BU method - $X_{10}X_{11}$". The middle range includes "With BU method - $X_{10}X_{12}$"and "Without BU method," indicating that the baseline method without BU outperforms some BU methods. Lower ranks are seen for "With BU method - X_{10}", "With BU method - X_{12}," and "With BU method - X_{11}," showing less effectiveness when each variable is used

Fig. 11. Rank for TOPSIS approach for the Heather Exchanger problem.

Table 7
Results of different network metrics on the G01 problem.

Method	Without BU method	With BU method						
		X_{10}	X_{11}	X_{12}	$X_{10}X_{11}$	$X_{10}X_{12}$	$X_{11}X_{12}$	$X_{10}X_{11}X_{12}$
Average Betweenness Centrality	0.045	0.055	0.052	0.053	0.052	0.052	0.052	0.0
Average Closeness Centrality	4.267e-12	370.195	6.260	74.975	1.656	0.889	0.844	0.0
Average Eigenvector Centrality	0.156	0.121	0.109	0.117	0.205	0.205	0.206	0.277
Average PageRank Centrality	0.076	0.076	0.076	0.076	0.076	0.076	0.076	0.076
Network Density	0.384	0.269	0.128	0.269	0.128	0.128	0.128	0
Average Clustering Coefficient	0.116	0.012	0.0	0.030	0.0	0.0	0.0	0
Average Weighted Degree	1.717e+12	0.160	0.156	0.177	0.392	0.729	0.764	0

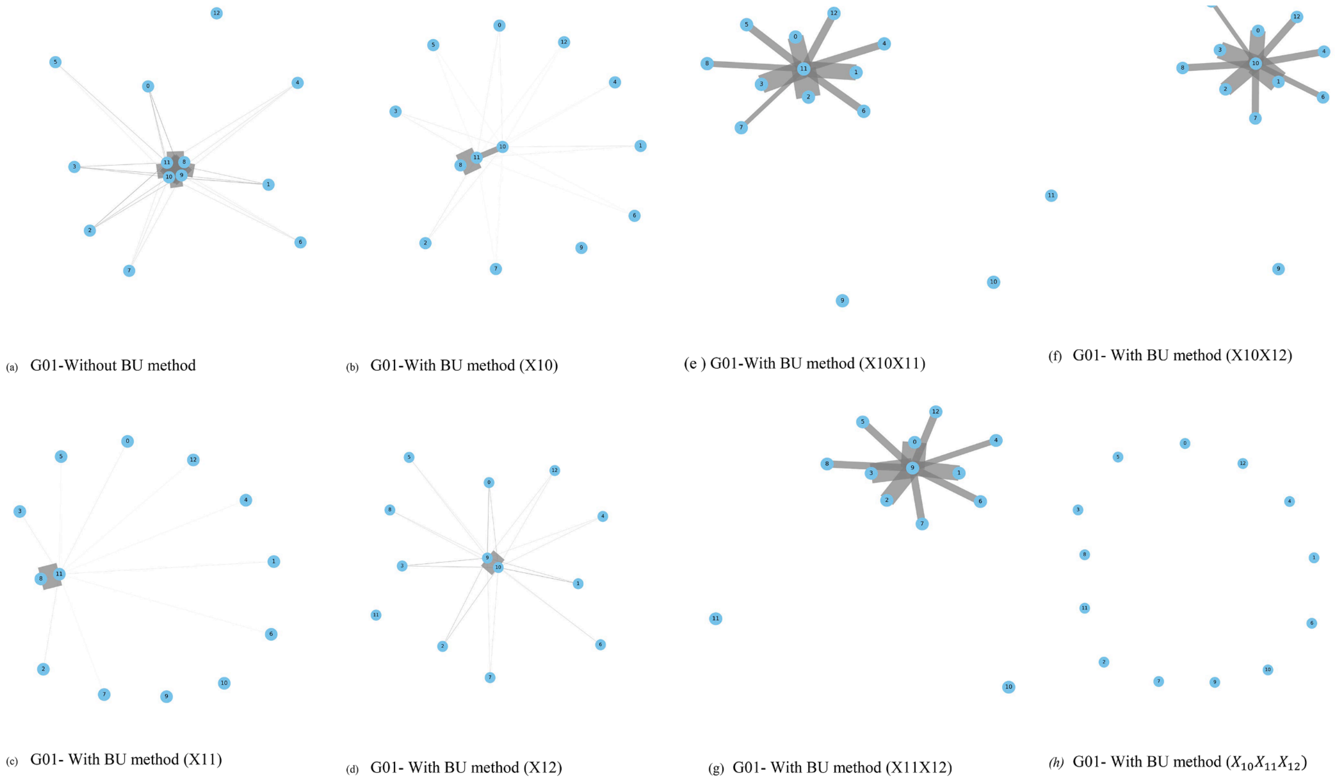


Fig. 12. Network visualization of different strategies on the G01 problem.

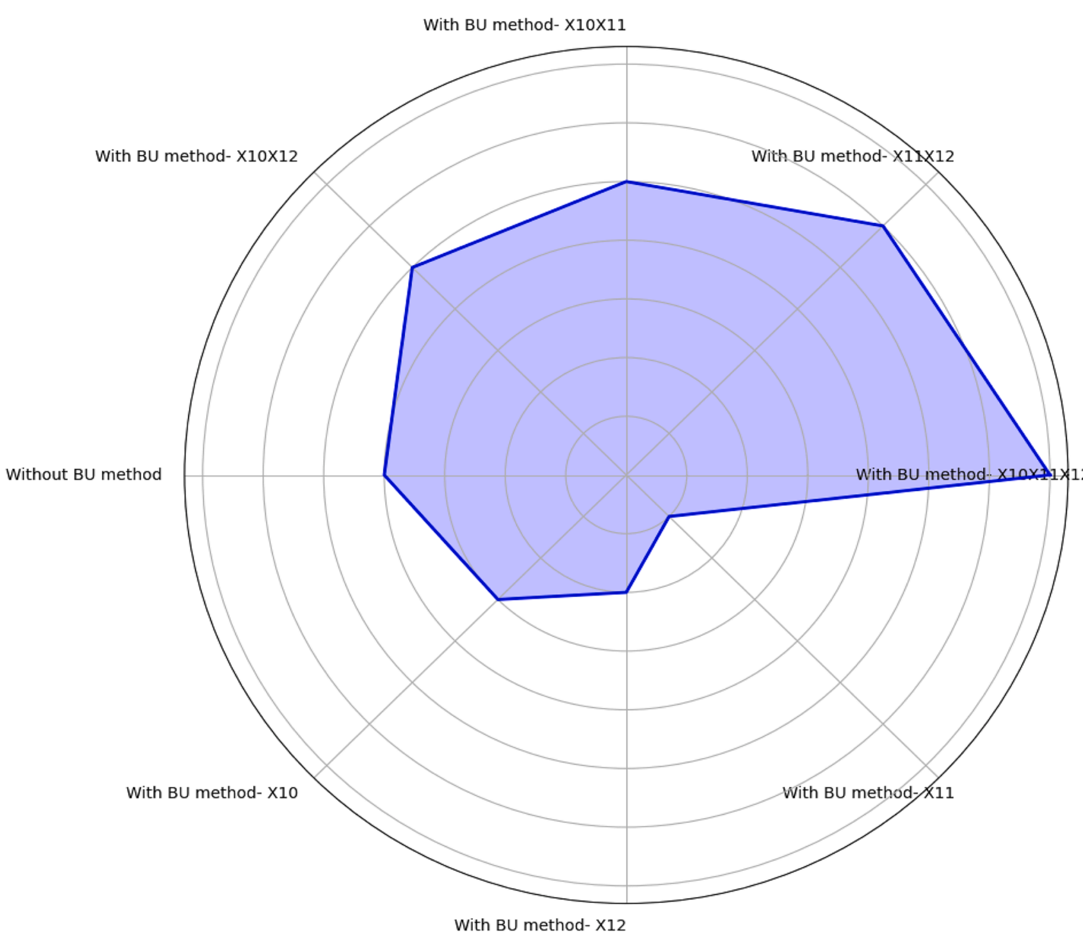


Fig. 13. Rank for TOPSIS approach for G01 problem.

individually in the BU method. This analysis underscores the superior performance of comprehensive variable integration in the BU method.

8. Discussion

The findings of this research show significant insights into the BU method's effectiveness when applied to constrained optimization problems. The detailed analysis of variable interactions using the DG2 algorithm and various network analysis metrics has shed light on the structural properties of optimization problems under different formulations. Several key points emerge from the discussion as follows:

- Impact of BU method: Applying the BU method can lead to less variable interaction and, therefore, a more simplified optimization landscape, which results in a reduction in infeasible search space. This reduction and simplification in search space complexity was evident across all benchmark problems, resulting in more independent variables and fewer interactions among them.
- Variable interaction analysis: The DG2 algorithm was used to identify the interactions between variables in both BU and without BU scenarios. The metrics, such as average betweenness centrality, closeness centrality, eigenvector centrality, and network density, provided comprehensive insights into the variable interactions. The BU method generally resulted in a more efficient network by analyzing the lower betweenness and closeness centrality values, which indicates fewer critical intermediary nodes and shorter paths between variables.
- MCDM Approach: The superiority of the BU method is further validated using the TOPSIS approach. The TOPSIS analysis ranked various BU strategies according to their proximity to an ideal solution, and the strategies involve multiple repairing variables that consistently rank higher. This method highlighted the robustness and reliability of the BU method, which handles complex constraints effectively.
- Trade-offs and Balance: A critical aspect of this study is finding the balance between shrinking the search space and managing the complexity of variable interactions. The BU method's ability to twist the search space while maintaining an efficient optimization process was observed in the findings. This balance is essential for ensuring that the performance gains from the BU method are not offset by increased problem complexity.
- The BU method improves optimization efficiency by dynamically adjusting the variable bounds during the optimization process, resulting in search space reduction. Specifically, the method identifies the constraints most critical to defining the feasible region and uses these constraints to iteratively update the variable boundaries. The BU method removes significant areas of the search space that are unlikely to contain optimal answers by pushing the search algorithm to focus on feasible regions by limiting the variable ranges. The idea of semi-independent variables guides this procedure, guaranteeing that the restrictions are handled in a way that reduces variable conflict and maintains the fundamental structure of the search space.
- The trade-off between the reduction of search space and the quality of solutions will be evaluated by incorporating variable interaction analysis through the DG2 algorithm. The method makes sure that narrowing the search space doesn't unnecessarily complicate the problem or make it impossible to find high-quality answers by assessing the interdependencies between variables. Furthermore, a multi-criteria decision-making methodology, TOPSIS, which rates the efficacy of various methods according to their proximity to an ideal solution, is used to validate the method's performance. This approach guarantees that the BU approach keeps efficiency and solution quality under check, as evidenced by its higher rankings across benchmark problems than without BU solutions.
- The BU method is particularly effective in optimization problems characterized by high-dimensional variable space with well-defined constraints. Examples of these optimization types include Welded Beam design problems, Pressure Vessel, and Spring Design Problems, where the constraints are critical and separable. The BU method can perform well in scenarios where reducing the infeasible search space can significantly streamline the optimization process and simplify variable interactions, resulting in algorithm pushes to focus on feasible regions. However, the BU method should be more explored in situations where constraints are loosely defined, highly dynamic, or interdependent to the extent that variable boundaries cannot be effectively isolated. For example, the assumption of semi-independent variables would not hold true in optimization situations with highly nonlinear, interdependent constraints, which could limit the method's efficacy.

9. Conclusion

This study endeavours to implement the BU method across various benchmark problems. The BU method enhances the optimization search by eliminating the infeasible search space; by applying this approach, the search area is twisted, and empirical results on this problem suggest that the BU method leads to a simpler variable interaction structure, which has positive impacts on the optimization algorithm's performance. This is because of the shrinkage of the search area due to the BU method and the reduction of the complexity of the optimization problem. Moreover, the reduction of the search space due to the BU method, the complexity of the optimization problem, and the search behaviour of different optimizers may possess different responses to different BU formulations. In this research, various formulations of the BU method are conducted, and the DG2 algorithm analyses the interaction of decision variables.

The analysis for the studied numerical examples, supported by various SNA metrics, illustrates that strategies with the highest number of repairing variables outperform other strategies. This superior performance presences more independent decision variables and fewer interdependencies among them, facilitating the discovery of optimal solutions.

Furthermore, the application of the MCDM approach, TOPSIS, enables a robust evaluation and ranking of different boundary update formulations. The results show that the boundary update method simplifies variable interactions and improves the optimization algorithm's overall effectiveness, resulting in the best optimal values as initially obtained [6].

Considering the advantages of the BU method, the proposed approach has several limitations. First of all, the iterative boundary adjustments and the incorporating of DG2 for variable interaction analysis result in increased processing cost. For large-scale size problems involving high-dimensional variable spaces, this overhead may become substantial. Future studies will investigate the usage of parallelized implementations or more effective algorithms to overcome this issue and cut down on calculation time. Also, the performance of the BU technique on real-world problems with extremely complicated constraints and dynamic environments is not entirely explored in the current study, which is restricted to benchmark tasks. The evaluation must be extended to real-world applications, including dynamic scheduling difficulties or industrial optimization tasks, to confirm the method's scalability and resilience. Furthermore, the BU method's reliance on semi-independent variables assumes a certain level of variable separability, which might not hold true for all optimization problem formulations. It could be required to adapt the BU method or hybrid approaches that incorporate other constraint-handling strategies for optimization issues with highly interdependent variables.

CRediT authorship contribution statement

Iman Rahimi: Writing – review & editing, Writing – original draft, Visualization, Investigation, Formal analysis, Data curation, Conceptualization. **Navid Yazdanjue:** Writing – review & editing, Visualization. **Mohammad Sadegh Khorshidi:** Writing – review & editing, Visualization. **Mohammad Reza Nikoo:** Writing – review & editing, Visualization, Validation. **Fang Chen:** Supervision. **Amir H. Gandomi:** Validation, Supervision, Project administration, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] R. Behmanesh, I. Rahimi, A.H. Gandomi, Evolutionary many-objective algorithms for combinatorial optimization problems: a comparative study, Archives of Computat. Methods Eng 28 (2) (2021) 673–688.
[2] C.A.C Coello, Multi-objective Optimization, 2018.
[3] K. Deb, Introduction to evolutionary multiobjective optimization, Multiobj. Optimiz.: Interac. Evolut. Appr. (2008) 59–96.
[4] F. Dahan, M.K. Roberts, M. Nagabushanam, T.M. Alfakih, An inspired chaos-based estimation-theory optimization for low-density parity-check (LDPC) code decoding, Resul. Eng 22 (2024) 101986.
[5] M.F. Javed, B. Siddiq, K. Onyelowe, W.A. Khan, M. Khan, Metaheuristic optimization algorithms-based prediction modeling for titanium dioxide-Assisted photocatalytic degradation of air contaminants, Resul. Eng 23 (2024) 102637.
[6] A.H. Gandomi, K. Deb, Implicit constraints handling for efficient search of feasible solutions, Comput. Methods Appl. Mech. Eng 363 (2020) 112917.
[7] A.H. Gandomi, K. Deb, R.C. Averill, S. Rahnamayan, M.N. Omidvar, Using semi-independent variables to enhance optimization search, Exp. Syst. Appl 120 (2019) 279–297.
[8] Z. Michalewicz, Genetic algorithms, numerical optimization, and constraints, Proceed. Sixth Int. Confer. Genet. Algo 195 (1995) 151–158.
[9] J. Mirabel, F. Lamiraux, Handling implicit and explicit constraints in manipulation planning, Robot.: Sci. Sys. 2018 (2018) 9p.
[10] M.N. Omidvar, X. Li, Y. Mei, X. Yao, Cooperative co-evolution with differential grouping for large scale optimization, IEEE Transac. Evolution. Comput 18 (3) (2013) 378–393.
[11] M.N. Omidvar, M. Yang, Y. Mei, X. Li, X. Yao, DG2: a faster and more accurate differential grouping for large-scale black-box optimization, IEEE Transac. Evolut. Comput 21 (6) (2017) 929–942.
[12] B. Raghavan, M. Xiao, P. Breitkopf, P. Villon, Implicit constraint handling for shape Optimization with pod-morphing, Eur. J. Computat. Mech 21 (3–6) (2012) 325–336.
[13] I. Rahimi, A.H. Gandomi, F. Chen, E. Mezura-Montes, A review on constraint handling techniques for population-based algorithms: from single-objective to multi-objective optimization, Arch. Computat. Methods Eng 30 (3) (2023) 2181–2209.
[14] I. Rahimi, A.H. Gandomi, M.R. Nikoo, M. Mousavi, F. Chen, Efficient implicit constraint handling approaches for constrained optimization problems, Sci Rep 14 (1) (2024) 4816.
[15] S.S. Rao, Engineering optimization: Theory and Practice, John Wiley & Sons, 2019.
[16] H. Rezk, A.G. Olabi, T. Wilberforce, E.T. Sayed, Metaheuristic optimization algorithms for real-world electrical and civil engineering application: a Review, Resul. Eng (2024) 102437.
[17] Sandgren, E. (1990). *Nonlinear integer and discrete programming in mechanical design optimization*.
[18] K. Uemura, N. Nakashima, Y. Nagata, I. Ono, A new real-coded genetic algorithm for implicit constrained black-box function optimization, 2013 IEEE Congr. Evolut. Computat (2013) 2887–2894.
[19] Wang, J., Westermann, R., Gao, X., & Wu, J. (2024). Design and optimization of functionally-graded triangular lattices for multiple loading conditions. *ArXiv Preprint ArXiv:2402.15458*.
[20] W. Wang, K. Wu, F. van Keulen, J. Wu, Regularization in space–time topology optimization for additive manufacturing, Comput. Methods Appl. Mech. Eng 431 (2024) 117202.
[21] B.D. Youn, K.K. Choi, A new response surface methodology for reliability-based design optimization, Comput. Struct 82 (2–3) (2004) 241–256.
[22] B.D. Youn, K.K. Choi, A new response surface methodology for reliability-based design optimization, Comput. Struct 82 (2–3) (2004) 241–256.