REST Journal on Advances in Mechanical Engineering

Vol: 3(2), June 2024

REST Publisher; ISSN: 2583-4800 (Online)

Website: https://restpublisher.com/journals/jame/

DOI: https://doi.org/10.46632/jame/3/2/3

Exploring Optimization Techniques: Applications in Metal Cutting, Food Chemistry, and Embedded Systems

*** [1]Somesh Nagalla, [1]Deepak Das, [2]Nihari Paladugu**

[1]IEEE Member, Texas, USA.

[2]IEEE Member, Ohio, USA.

**Corresponding Author Email: somesh.nagalla@gmail.com*

Abstract. *The discussion further extends to various fields such as maintenance scheduling, memory energy consumption in processor-based architectures, multivariate optimization processes, and query optimization methods. The paper explores the use of optimization techniques in food analytical chemistry explores geometric programming and discusses optimization of control systems using methods inspired by nature. Research significance: A critical evaluation of current techniques for modelling and optimizing input-output and process parameters in metal cutting processes identifies key issues that need to be addressed for effective optimization, providing valuable insights to researchers and practitioners in the field. Furthermore, providing a broad framework for improving metal cutting processes is a valuable contribution to guide future research and development efforts. Mythology: Alternative: Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Simulated Annealing (SA), Ant Colony Optimization (ACO), Differential Evolution (DE)Evaluation Preference: Efficiency (%), Cost Savings (%), Execution Time (minutes), Resource Consumption (units) Result: The results indicate that Resource Consumption (units) achieved the highest rank, while Simulated Annealing (SA) had the lowest rank being attained. Conclusion: "The value of the dataset for Resource Consumption (units), according to the weighted sum method, Fibre-Reinforced Polymer (FRP) Composites achieves the highest ranking*

Key words: *Maintenance Scheduling Optimization, Memory Power Consumption, Multivariate Optimization, Query Optimization, Food Analytical Chemistry Optimization, Geometric Programming,*

1. INTRODUCTION

This paper aims to comprehensively evaluate current techniques for modelling and optimizing linkages between input-output and process parameters in metal cutting operations. It is recommended to use the detailed framework to carry out process optimization studies in metal cutting. [1] None of the previously mentioned literature reviews cover efficiency optimization of winds turbines. Therefore, this paper aims to review the optimization techniques used specifically for wind turbines. [3] We explore modern optimization techniques and their application to optimization challenges in integrated circuit design. This review covers the theory, methods, and programs associated with these techniques and assesses their current and future impact on integrated circuit design. Integrated circuits must handle complex trade-offs between various linear objectives and constraints, which are often nonlinear and rarely converge. Estimating functions and gradients involves solving a large number of nonlinear differential equations, which can be imprecise and expensive. In addition, the parameters to be optimized are affected by inherent statistical variability. Our focus is on multi-objective constrained optimization techniques suitable for these conditions. [4] We will also provide a brief summary of recent advances in simulation optimization for discrete parameters. However, this discussion does not cover related methods such as ranking, selection, and multiple comparison techniques, solutions to the multiple armed robbery problem, or learning automated processes. [5] A key method uses ranking techniques or comparable strategies. The COMOGA method is presented as a solution that treats each constraint separately. [7] Historically, in-memory system design has focused on the compiler, architecture, and CAD domains. Although many of these methods are valuable, they do not fully exploit the optimization opportunities that exist in embedded system design. Purpose-built embedded systems can benefit from advanced optimization techniques that take full advantage of application-specific insights. In contrast to conventional memory hardware and software upgrades, which must handle the variability of general-purpose applications, memory upgrades for embedded systems can be customized to meet the specific needs of their code and data. Furthermore, embedded system designers are focusing on designing memory subsystems both on-chip and off-chip, resulting in non-traditional memory architectures, with static cache hierarchy as one of the many architectural possibilities. [9] The approaches to be studied differ greatly in how they define the optimality criteria and in the optimization techniques used. Typically, a unit should be taken offline for maintenance once or twice a year, for one to a few weeks. This maintenance should normally happen within a specific time frame for each unit. Maintenance resources such as MW, parts and manpower are running low every week. Additionally, there may be restrictions on the order in which units are maintained, such as allowing one unit from a particular group to be offline at a time. Depending on the decision variables that determine the maintenance schedule for each unit, these constraints are classified as linear equality or inequality constraints. [11] With a fixed memory hierarchy, the only way to increase memory power consumption is to change the memory access patterns required for computations. In processor-based architectures, the source code optimization techniques described in Section 4.2 are used. At more abstract levels, memory power consumption can be reduced by carefully selecting data structures beyond the source code. [13] Since many decisions depend on the operator in the optimization process, such as selecting variables, determining their test domains, and fully evaluating the results, it is necessary to engage with "good knowledge."Furthermore, all data generated by statistical programs must be critically evaluated, underscoring the importance of human involvement in multivariate optimization processes. [14]This approach is more efficient compared to the exponential join counting methods used by traditional query optimizers. Additionally, if no expensive predictions are required during query parsing, the discussed methods can be omitted. For queries with expensive predicates, the gains in execution speed are expected to outweigh the additional time spent on optimization. [17]Chemical instruments can be used in food analytical chemistry in two main ways: including selecting variables, defining their test domains, and carefully evaluating outcomes, engaging with "good knowledge" is critical. especially for robustness testing. [18] Regarding the theory of geometric programming and its equivalence to fixed point conditions, Hall proved this equivalence in his optimization model. In contrast, we make a straightforward application of the Kuhn–Tucker theorem to both of our models. Although Hall recommends solving the nonlinear model by gradient descent, we recommend using alternative techniques that use existing efficient mathematical programming tools for linear network problems.[19] These techniques improve problem-solving skills by modifying existing methods and combining different approaches to create new ones. For example, differential search algorithms are used to improve fuzzy logic controller (FLC) design and control of photovoltaic (PV) inverters. [20] David Ann's method is widely used for various problems and has demonstrated excellent performance. It is considered one of the most effective optimization techniques today. However, there is increasing interest in improving methods for optimizing the matrix H. [21] this paper seeks to provide a summary of recent research on the optimal design and installation of photovoltaic (PV) power systems. The second section focuses on techniques for optimizing standalone photovoltaic (PV) systems. The third, fourth, and fifth sections explore optimization techniques for PV/diesel generator systems, PV/wind systems, and grid-connected systems, respectively. The sixth section explores optimization methods for sizing inverters in photovoltaic (PV) systems. tickle seven concludes with a discussion of challenges in scaling photovoltaic (PV) systems. [22].

2. MATERIALS AND METHODS

Recently, the weighted sum method has received considerable attention as a practical tool, with an extensive literature describing its applications. However, most of this attention has been on practical applications, often solving problems with two objective functions. For example, coo gee and Silvenoinen (1987) showed an early A weighted sum method is used by systematically adjusting the weights to find various Pareto optimal solutions. Their approach was used to minimize both the volume and nodal displacement of a four-bar span truss, demonstrating the effectiveness of the technique in multi-objective optimization. The weighted product method is similar to the weighted sum method except that it uses multiplication instead of addition. This approach involves either pre-defining the weight vector or adjusting it incrementally during the search process. In multi-objective structural optimization, a weighted sum method is used to achieve multiple Pareto optimal solutions by applying predefined weights and systematically modifying these weights through a series of algorithm iterations. Likewise, it has been used for topology optimization in other studies. The adaptive weight sum method generates a heterogeneous and well-distributed set of solutions. It effectively identifies Pareto optimal solutions in dense regions while also recognizing non-Pareto optimal solutions. First, the constant weighted sum method is used to estimate the shape of the Pareto front. At each iteration, a meta-model is generated for each individual objective function. The Confidence Zone (CON) method is used to establish the sample area for these meta-models. [8] The sample size of individual studies is considered the optimal weight for the method. Although the detailed characteristics of each method and the analytical relationships between them have not been fully explored, empirical evidence indicates that the two methods perform similarly. [12] In the first step, a constant weighted sum method is used to quickly approximate the general shape of the Pareto front and establish a network of links. Each link adds additional constraints to indicate potential areas for deeper exploration. In the next step, a weighted sum method is used within these feasible regions to find additional Pareto optimal solutions. By incorporating additional inequality constraints in the constant weighted sum method, the optimization process is directed toward newly defined areas that require further investigation. The Adaptive Weighted Sum (AWS) method effectively tackles multi-objective optimization problems by generating well-distributed solutions, finding Pareto optimal solutions in convergent regions, and differentiating non-Pareto optimal solutions. Previously, the AWS method was limited to bi-objective optimization problems. We evaluated the accuracy of the weighted sum method for calculating ground forces. To find the optimal experimental conditions, we optimized the variables such as volume of sprayed wastewater, temperature, carbonation period and pre-drying method using the linear weighted sum method. The amount of surface applied wastewater varies from 20% to 80% of the total water absorption. [16].

3. ANALYSIS AND DISCUSSION

TABLE 1. Optimization Techniques

	Efficiency (%)	Cost Savings (%)	Execution Time (minutes)	Resource Consumption (units)
Genetic Algorithm (GA)	85	20	15	10
Particle Swarm Optimization (PSO)	78	25	12	8
Simulated Annealing (SA)	90	18	20	15
Ant Colony Optimization (ACO)	75	30	10	12
Differential Evolution (DE)	80	22	18	9

Efficiency and cost savings: Simulated Annealing (SA) has a maximum efficiency of 90%, indicating its best performance in finding optimal solutions. However, it offers a savings of 18% at a lower cost. On the other hand, Ant Colony Optimization (ACO) has high cost savings of 30% but low efficiency of 75%. Genetic algorithm (GA), particle swarm optimization (PSO) and differential evolution (DE) demonstrate moderate efficiency and cost savings. Specifically, GA achieves 85% efficiency and 20% cost savings, PSO achieves 78% efficiency and 25% cost savings, and DE provides 80% efficiency and 22% cost savings. Regarding execution time and resource consumption, Ant Colony Optimization (ACO) is faster, completing tasks in 10 minutes, while PSO is completed in 12 minutes. Simulated annealing (SA) takes longer at 20 min. Resource consumption varies, with GA and SA consuming the most resources at 10 and 15 units, respectively, while PSO consumes the least at 8 units.

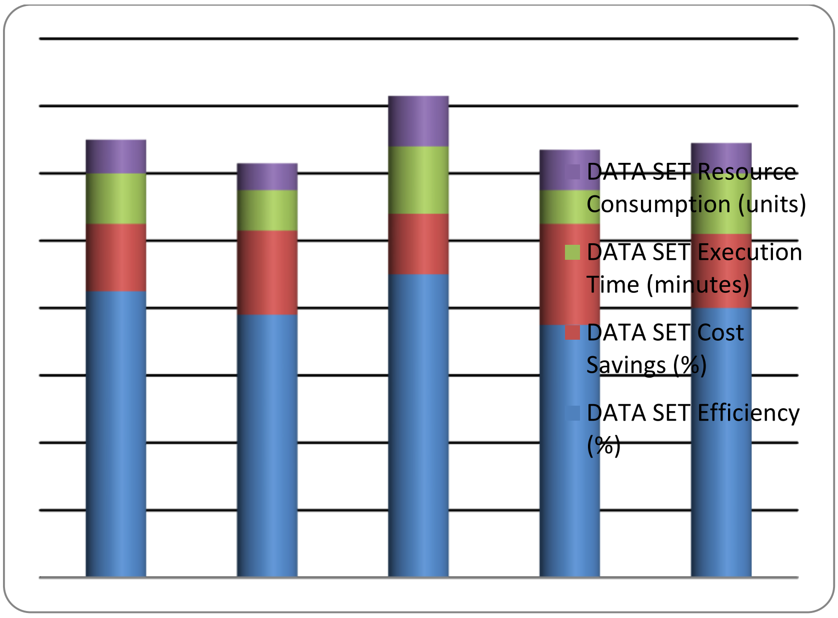

FIGURE 1. Optimization Techniques

Bar chart contrasts the performance of five optimization algorithms—genetic algorithm (GA), particle swarm optimization (PSO), simulated annealing (SA), ant colony optimization (ACO) and differential evolution (DE)—on four metrics: resource consumption, execution time, cost Savings and efficiency. Each bar is divided into four sections representing each metric. The largest section for each algorithm is "DATA SET Efficiency (%)" (Shown in blue), indicating that efficiency is an important factor in the overall comparison. GA shows moderate performance but has high resource consumption compared to others. PSO has relatively balanced performance in all metrics, moderate resource consumption and processing time, and excellent cost savings. SA shows the highest processing time (green), indicating that the algorithms take longer to run. ACO shows strong balance with good performance and low operation time. However, its cost savings and resource consumption are modest. DE, on the other hand, has slightly better performance than ACO, but with the same level of resource consumption and execution time.

TABLE 2. Normalized

Normalized			
0.94444	0.66667	0.66667	0.80000
0.86667	0.83333	0.83333	1.00000
1.00000	0.60000	0.50000	0.53333
0.83333	1.00000	1.00000	0.66667
0.88889	0.73333	0.55556	0.88889

Best performance: The second row with normalized values of 0.86667, 0.83333, 0.83333 and 1.00000 indicates a consistent high performance algorithm in all metrics. A maximum value of 1.00000 indicates optimal resource consumption, making it a well-rounded choice. Similarly, the fourth row scores a perfect 1.00000 in both cost savings and execution time, suggesting that the algorithm is exceptionally robust in these areas, even though it has low values for efficiency and resource consumption. Moderate performance: The first row shows normalized values of 0.94444, 0.66667, 0.66667 and 0.80000, indicating strong performance but moderate cost savings and processing time. The fifth row shows a balanced performance, with values of 0.88889 for efficiency and resource consumption, but lower for cost savings and execution time.

TABLE 3. Weighted normalized decision matrix

Weighted normalized decision matrix			
0.23611	0.16667	0.16667	0.20000
0.21667	0.20833	0.20833	0.25000
0.25000	0.15000	0.12500	0.13333
0.20833	0.25000	0.25000	0.16667
0.22222	0.18333	0.13889	0.22222

Weighted scores: The second and fourth rows stand out, with relatively heavy values of 0.20833, 0.20833, 0.25000 (second row) and 0.20833, 0.25000, 0.25000 (fourth row, calculating efficiency and specific time). These sequences suggest that these mechanisms may be highly balanced, providing a good trade-off between various factors such as performance, cost, and resource consumption. Moderate scores: In the first row, the values 0.23611, 0.16667, 0.16667 and 0.20000 show high scores for the first criterion, but moderate scores for the others. This suggests that although this algorithm is highly efficient, it may not perform well in terms of cost savings and execution time. Low-weighted scores: The third row has low-weighted scores, especially for processing time and resource consumption (0.12500 and 0.13333), although the first criterion has a high score. This represents a trade-off where some algorithms excel in specific areas but are less balanced on all important criteria. The fifth row exhibits a more balanced but slightly lowers overall performance across all criteria.

TABLE 4. Preference Score

Preference Score	
Genetic Algorithm (GA)	0.76944
Particle Swarm Optimization (PSO)	0.88333
Simulated Annealing (SA)	0.65833
Ant Colony Optimization (ACO)	0.87500
Differential Evolution (DE)	0.76667

Top Preferences: Particle Swarm Optimization (PSO) has the highest preference score of 0.88333, suggesting that it is the most preferred algorithm due to its strong balance across all criteria. Ant Colony Optimization (ACO) is far behind with a score of 0.87500, indicating that it is very useful, especially in situations where cost savings and quick processing are important. Mid-tier preferences: Genetic Algorithm (GA) and Differential Evolution (DE) have preference scores of 0.76944 and 0.76667, respectively. These scores reflect moderate performance across all criteria, making them reliable choices depending on specific problem requirements, but not outperforming the best ranking algorithms. Low Priority: Simulated Annealing (SA) has the lowest priority score at 0.65833. While it excels in some areas such as efficiency, it is less efficient in terms of cost savings, processing time or resource consumption, implying that it is overall less favourable compared to other methods.

TABLE 5. Rank

Rank	
Genetic Algorithm (GA)	3
Particle Swarm Optimization (PSO)	1
Simulated Annealing (SA)	5
Ant Colony Optimization (ACO)	2
Differential Evolution (DE)	4

Top Ranked Algorithms: Particle Swarm Optimization (PSO) ranks first, which is the most effective algorithm in balancing multiple performance criteria such as efficiency, cost savings, processing time, and resource consumption. Its high ranking indicates a strong ability to effectively solve optimization problems. Ant Colony Optimization (ACO) is a strong contender that offers high performance, especially when cost savings and fast processing times are prioritized. In the middle tier, the Genetic Algorithm (GA) ranks third, showing solid performance across different parameters. It offers a strong balance of performance and compatibility, making it a reliable option for various optimization tasks. Differential evolution (DE) is ranked fourth, indicating that while useful, it may not perform better than the best algorithms in some situations.

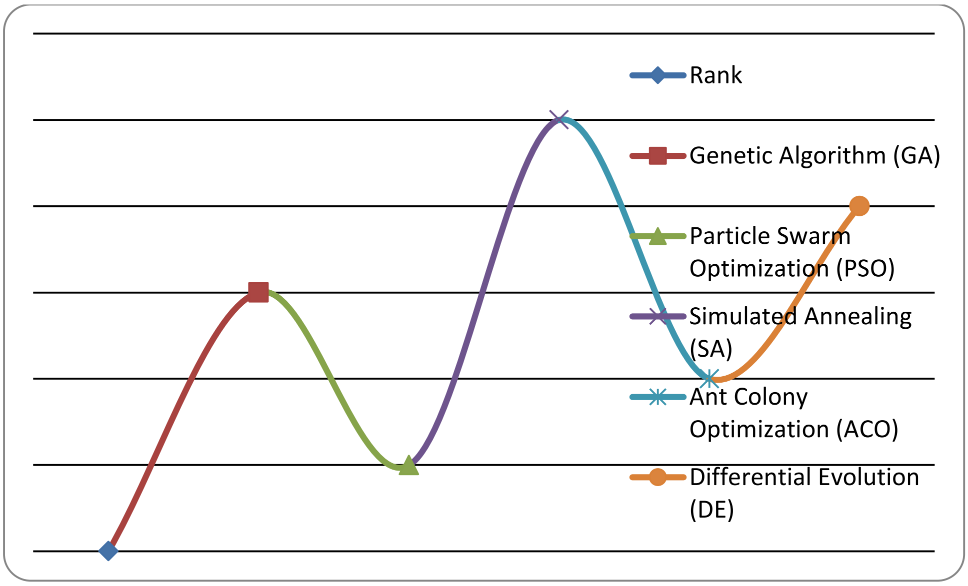

FIGURE 2. Rank

The line chart shows the ranking of five optimization algorithms—genetic algorithm (GA), particle swarm optimization (PSO), simulated annealing (SA), ant colony optimization (ACO), and differential evolution (DE)—based on a specific performance measure. , denoted by "rank" on the y-axis. From the table, Genetic Algorithm (GA) starts with relatively low quality, indicating strong performance compared to others. The ranking rises sharply with Particle Swarm Optimization (PSO), which shows less favourable performance. Simulated Annealing (SA) achieves the highest quality value in the chart, suggesting that it is the least efficient or effective of the algorithms in this particular measurement. Following this, Ant Colony Optimization (ACO) shows significant improvement, indicated by a drop in rank, reflecting better performance. Differential evolution (DE) is slightly superior to ACO but less than GA.

4. CONCLUSION

This paper critically evaluates existing techniques for modelling and optimizing input-output relationships and process parameters in metal cutting processes. It identifies key issues that need to be addressed for effective optimization of these parameters. A general framework for process optimization studies related to metal cutting is proposed, providing a structured approach to address the identified challenges. The review covers a variety of optimization methods, including wind turbines, integrated circuits, embedded systems, and PV systems, highlighting the diversity in optimization challenges in different fields. In addition, the paper explores advanced optimization techniques such as the adaptive weighted sum method, which has proven useful in multi-objective optimization scenarios, but has limitations in its applicability to problems with more than two objectives. The findings emphasize the importance of combining different techniques and designing strategies optimized for the specific characteristics of each application to achieve optimal outcomes.

REFERENCES

[1]. Mukherjee, Indrajit, and Pradip Kumar Ray. "A review of optimization techniques in metal cutting processes." Computers & Industrial Engineering 50, no. 1-2 (2006): 15-34.

[2]. Chehouri, Adam, Rafic Younes, Adrian Ilinca, and Jean Perron. "Review of performance optimization techniques applied to wind turbines." Applied Energy 142 (2015): 361-388.

[3]. Brayton, Robert K., Gary D. Hachtel, and Alberto L. Sangiovanni-Vincentelli. "A survey of optimization techniques for integrated-circuit design." Proceedings of the IEEE 69, no. 10 (1981): 1334-1362.

[4]. Andradóttir, Sigrún. "A review of simulation optimization techniques." In 1998 winter simulation conference. Proceedings (Cat. No. 98CH36274), vol. 1, pp. 151-158. IEEE, 1998.

[5]. Coello, Carlos A. "An updated survey of GA-based multiobjective optimization techniques." ACM Computing Surveys (CSUR) 32, no. 2 (2000): 109-143.

[6]. Panda, Preeti Ranjan, Francky Catthoor, Nikil D. Dutt, Koen Danckaert, Erik Brockmeyer, Chidamber Kulkarni, A. Vandercappelle, and Per Gunnar Kjeldsberg. "Data and memory optimization techniques for embedded systems." ACM Transactions on Design Automation of Electronic Systems (TODAES) 6, no. 2 (2001): 149-206.

[7]. Sasson, Albert M., and HYDE M. Merrill. "Some applications of optimization techniques to power systems problems." Proceedings of the IEEE 62, no. 7 (1974): 959-972.

[8]. Benini, Luca, and Giovanni de Micheli. "System-level power optimization: techniques and tools." ACM Transactions on Design Automation of Electronic Systems (TODAES) 5, no. 2 (2000): 115-192.

[9]. Ferreira, Sergio LC, Valfredo A. Lemos, Vanessa S. de Carvalho, Erik GP da Silva, Antonio FS Queiroz, Caio SA Felix, Daniel LF da Silva, Gilson B. Dourado, and Rafael V. Oliveira. "Multivariate optimization techniques in analytical chemistry-an overview." Microchemical Journal 140 (2018): 176-182.

[10]. Hellerstein, Joseph M. "Optimization techniques for queries with expensive methods." ACM Transactions on Database Systems (TODS) 23, no. 2 (1998): 113-157.

[11]. Ferreira, Sergio LC, Mario M. Silva Junior, Caio SA Felix, Daniel LF da Silva, Adilson S. Santos, João H. Santos Neto, Cheilane T. de Souza, Raineldes A. Cruz Junior, and Anderson S. Souza. "Multivariate optimization techniques in food analysis–A review." Food chemistry 273 (2019): 3-8.

[12]. Collins, M., Leon Cooper, Richard Helgason, Jeffery Kennington, and Larry LeBlanc. "Solving the pipe network analysis problem using optimization techniques." Management science 24, no. 7 (1978): 747-760.

[13]. Hannan, M. A., Jamal A. Ali, Azah Mohamed, and Aini Hussain. "Optimization techniques to enhance the performance of induction motor drives: A review." Renewable and Sustainable Energy Reviews 81 (2018): 1611-1626.

[14]. Swann, W. H. "A survey of non-linear optimization techniques." FEBS letters 2, no. S1 (1969): S39-S55.

[15]. Khatib, Tamer, Azah Mohamed, and Kamaruzzaman Sopian. "A review of photovoltaic systems size optimization techniques." Renewable and Sustainable Energy Reviews 22 (2013): 454-465.

[16]. Marler, R. Timothy, and Jasbir S. Arora. "The weighted sum method for multi-objective optimization: new insights." Structural and multidisciplinary optimization 41 (2010): 853-862.

[17]. San Cristóbal Mateo, José Ramón, and José Ramón San Cristóbal Mateo. "Weighted sum method and weighted product method." Multi criteria analysis in the renewable energy industry (2012): 19-22.

[18]. Wang, Rui, Zhongbao Zhou, Hisao Ishibuchi, Tianjun Liao, and Tao Zhang. "Localized weighted sum method for many-objective optimization." IEEE Transactions on Evolutionary Computation 22, no. 1 (2016): 3-18.

[19]. Kim, Il Yong, and Oliver L. De Weck. "Adaptive weighted-sum method for bi-objective optimization: Pareto front generation." Structural and multidisciplinary optimization 29 (2005): 149-158.

[20]. Kim, Il Yong, and Olivier L. de Weck. "Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation." Structural and multidisciplinary optimization 31, no. 2 (2006): 105-116.

[21]. Ryu, Jong-hyun, Sujin Kim, and Hong Wan. "Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization." In Proceedings of the 2009 Winter Simulation Conference (WSC), pp. 623-633. IEEE, 2009.

[22]. Lee, Cue Hyunkyu, Seungho Cook, Ji Sung Lee, and Buhm Han. "Comparison of two meta-analysis methods: inverse-variance-weighted average and weighted sum of Z-scores." Genomics & informatics 14, no. 4 (2016): 173.

[23]. Modest, Michael F. "The weighted-sum-of-gray-gases model for arbitrary solution methods in radiative transfer." (1991): 650-656.
[24]. Zhang, Ke-shi, Zhong-hua Han, Wei-ji Li, and Wen-ping Song. "Bilevel adaptive weighted sum method for multidisciplinary multi-objective optimization." AIAA journal 46, no. 10 (2008): 2611-2622.
[25]. Kim, Il Yong, and Olivier de Weck. "Adaptive weighted sum method for multiobjective optimization." In 10th AIAA/ISSMO multidisciplinary analysis and optimization conference, p. 4322. 2004.
[26]. Van Der Veen, Michiel, Jan Brouwer, and Klaus Helbig. "Weighted sum method for calculating ground force: an evaluation by using a portable vibrator system." Geophysical Prospecting 47, no. 3 (1999): 251-267.
[27]. Sojobi, Adebayo Olatunbsoun, Dongxing Xuan, Long Li, Songhui Liu, and Chi Sun Poon. "Optimization of gas-solid carbonation conditions of recycled aggregates using a linear weighted sum method." Developments in the Built Environment 7 (2021): 100053.