Journal of Network Intelligence ©2025 ISSN 2414-8105
Taiwan Ubiquitous Information Volume **10**, Number **4**, November **2025**

Product Design Structure Optimization Based on Improved SOM Clustering

Wang Ting*

School of Architectural Engineering
Xuzhou Vocational College of Industrial Technology
Xuzhou 221140, P. R. China
717483823@qq.com

Zhou Xiang Zhen

Faculty of Information Science and Technology
National University of Malaysia
Selangor 43600, Malaysia
619543699@qq.com

*Corresponding author: Wang Ting
Received January 29, 2024, revised February 3, 2025, accepted May 12, 2025.

Abstract. *This study aims to optimize product design structure by introducing enhanced self-organizing mapping (SOM) clustering methods to improve design efficiency and product performance. Traditional product design structure optimization is often constrained by high-dimensional data processing and complex relationship recognition. The enhanced SOM algorithm proposed in this paper considers the topological structure of traditional SOM and introduces adaptive learning rate and neighborhood function to enhance adaptability to high-dimensional data. The method initializes the information as the input layer of the SOM network, and then obtains the clustering scheme by training the SOM neural network model. The article also proposes clustering performance evaluation indicators that comprehensively consider module cohesion and coupling and are suitable for numerical DSM. In the experiment, we used multidimensional datasets from the field of product design for training and validation. By mapping design parameters to the SOM space, automatic clustering of design parameters was achieved, revealing potential relationships. Then, based on the clustering results, the product design structure was optimized. Finally, by analyzing the design characteristics of different cluster groups based on Adam, targeted improvements were made to the product structure, and the improved SOM algorithm effectively solved the challenges related to high-dimensional data and complex relationships. This method not only improves design efficiency, but also provides a feasible strategy for product design optimization in manufacturing and engineering. The performance score reached 90.8%, with an accuracy rate of 94.5%.*
Keywords: Self-Organizing Map (SOM); Clustering; Product Design; Structure Optimization; Enhanced SOM

---

1. Introduction.

Traditional approaches to product design structure optimization often grapple with challenges associated with the high dimensionality of design parameters and the intricate relationships among them. Recognizing the limitations of existing methodologies, this study seeks to pioneer a novel avenue for optimizing product design structures by leveraging an enhanced Self-Organizing Map (SOM) clustering method. Liao [1] proposed a multi-behavior RFM model based on improved SOM neural network algorithm for customer segmentation. Xia [2] made a rock burst intensity prediction in underground buildings based on improved spectral clustering algorithm. Wang [3] proposed an intrusion detection and performance simulation based on improved sequential pattern mining algorithm. Yang [4] proposed mobile sensors dispatch scheme based on improved SOM algorithm for coverage hole healing. However, these methods lack commonly used module clustering evaluation methods and tools, as there are many products module partitioning methods and diverse product design objectives. To address the nuanced requirements of product design, we propose an improved SOM algorithm that incorporates adaptive learning rates and sophisticated neighborhood functions. These enhancements are designed to augment the adaptability of the SOM model to intricate and high-dimensional data structures, facilitating a more nuanced understanding of the relationships among design parameters. Through empirical experiments employing real-world multidimensional datasets from the field of product design, we aim to demonstrate the efficacy of the proposed methodology in automatic design parameter clustering.

1.1. **Related work.** The field of product design structure optimization has witnessed a rich tapestry of research efforts aimed at enhancing methodologies and addressing the multifaceted challenges inherent in the design process. This section provides an overview of key literature related to both traditional approaches and recent advancements in the domain. Wang et al. [5] proposed a multi-attribute fusion algorithm based on improved evidence theory and clustering. Rong [6] made an analysis of the comprehensive evaluation model of enterprise technological innovation ability based on improved genetic algorithm. Xiao et al. [7] proposed a coal classification method based on improved local receptive field-based extreme learning machine algorithm. Yu et al. [8] proposed an identification method of abnormal characteristic data of power equipment based on improved K-Means algorithm. Ding et al. [9] made segmentation of the fabric pattern based on improved fruit fly optimization algorithm. Zhang et al. [10] studied equipment health assessment based on improved incremental support vector data description. Huo et al. [11] proposed a traffic anomaly detection method based on improved GRU and EFMS-K means clustering. Yan et al. [12] made prediction of bank telephone marketing results based on improved whale algorithms optimizing S-Kohonen network. Lu et al. [13] studied three-Dimensional path planning of deep-Sea mining vehicle based on improved particle swarm optimization. Guo et al. [14] made research on indoor wireless positioning precision optimization based on UWB. Ma et al. [15] proposed an unbalanced abnormal traffic detection based on improved Res-BIGRU and integrated dynamic ELM optimization.

The subsequent sections of this paper will delve into the methodology employed, detailing the process of training the enhanced SOM model and conducting clustering analysis on design parameters. We will elucidate the experimental setup, including the characteristics of the datasets used and the evaluation metrics applied to assess the performance of the proposed approach. The results and discussion section will showcase the outcomes of our experiments, shedding light on the implications of the enhanced SOM algorithm for product design structure optimization. Liu et al. [16] made research on innovative methods based on clustering and case-based reasoning. Wang et al. [17] studied efficient DDoS attack detection based on improved firefly algorithm to optimize convolutional neural networks. Zhao et al. [18] studied real-time detection of particleboard surface defects based on improved YOLOV5 target detection. Seghers et al. [19] studied unsupervised learning: local and global structure preservation in industrial data. Ma et al. [20] studied inversion of soil organic matter content based on improved convolutional neural network. Wang [21] made an efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach. Ran et al. [22] made an anomaly detection for hyperspectral images based on improved low-rank and sparse representation and joint Gaussian mixture distribution. Jiang and Liu [23] proposed an improved speech segmentation and clustering algorithm based on SOM and k-means. He et al. [24] proposed a new approach to modeling land use change. Cai et al. [25] made a difference Analysis of Regional Economic Development Based on the SOM Neural Network with the Hybrid Genetic Algorithm. Yan et al. [26] proposed a clustering algorithm for multi-modal heterogeneous big data with abnormal data. Liu and Li [27] coordinated obstacle avoidance of multi-AUV based on improved artificial potential field method and consistency protocol. Liu and Zhao [28] made research on an Improved SOM Model for Damage Identification of Concrete Structures. Tong et al. [29] made product development (PD) and service innovation (SI) to shape the upgraded model of SOM.

This study aspires to contribute to the evolving landscape of product design by introducing an innovative approach that not only addresses the challenges posed by high-dimensional data but also considers the intricate relationships governing design structures. The ultimate goal is to provide a robust and adaptable strategy for optimizing product design structures, thereby advancing the efficiency, performance, and viability of designed products in diverse industrial contexts.

1. **Traditional Approaches to Design Optimization.** Previous research in product design optimization has predominantly relied on mathematical modeling, simulation, and optimization algorithms. These methods often involve the formulation of mathematical models that represent the relationships between design parameters and desired performance metrics. Optimization algorithms, such as genetic algorithms and gradient-based approaches, are then employed to search for optimal design configurations within the defined parameter space. While effective in certain contexts, these approaches may struggle with the curse of dimensionality and may not fully capture the intricate interactions among design variables.
2. **Clustering in Product Design.** Clustering techniques have been employed to categorize design parameters and identify patterns within high-dimensional datasets. K-means clustering and hierarchical clustering are among the commonly used methods. However, these traditional clustering approaches may face challenges in handling complex and non-linear relationships among design variables.
3. **Self-Organizing Maps in Design Exploration.** The application of Self-Organizing Maps (SOM) in product design exploration has gained attention for its ability to uncover inherent structures in multidimensional data. SOM has been used for visualization and classification tasks, providing a holistic understanding of design spaces. However, traditional SOM algorithms may have limitations in adaptability and may struggle with non-uniform data distributions.
4. **Enhancements to Self-Organizing Maps.** Recent research has focused on enhancing the capabilities of SOM for improved adaptability to complex datasets. Adaptive learning rates and neighborhood functions have been introduced to address challenges related to high-dimensional and non-linear data. Growing Self-Organizing Maps (GSOM) and other variations aim to overcome limitations of traditional SOM, providing more nuanced representations of design spaces.
5. **Integration of Optimization and Clustering.** Some studies have explored the integration of optimization algorithms with clustering techniques. This synergy aims to leverage the strengths of both approaches, combining the global search capabilities of optimization algorithms with the pattern recognition capabilities of clustering methods. Such integrative approaches show promise in addressing the challenges of design space exploration.

In light of the aforementioned literature, our study contributes to this evolving landscape by proposing an enhanced SOM clustering method for product design structure optimization. By combining the capabilities of SOM with adaptive learning and sophisticated neighborhood functions, we aim to provide a more robust and adaptable solution to the challenges posed by high-dimensional and complex design datasets. The following sections will delve into the methodology, experiments, and results, offering a comprehensive exploration of the proposed approach and its implications for the field of product design optimization. The motivation behind this research stems from the pressing need to overcome the challenges and limitations inherent in traditional product design structure optimization methodologies. As the complexity of products and design parameters continues to increase, there is a growing recognition that existing approaches may struggle to provide comprehensive and efficient solutions. This study is motivated by a desire to push the boundaries of innovation in product design optimization and contribute to a more nuanced and adaptive framework.

2. Reconstruction models and common structures.

2.1. Reconstruction Models.

In the context of Self-Organizing Maps (SOM), a reconstruction model refers to the capability of the SOM to map input data onto its neurons and then reconstruct or represent that data. In the context of enhanced SOM, the main difference in SOM network structure lies in the competition layer, which is the number of SOM neurons in the competition layer, as shown in Fig. 1.
(1) Weight Vectors. Each neuron in the SOM has an associated weight vector. During training, the SOM adapts these weight vectors to approximate the input data distribution. An empirical formula for the minimum neuron count is

$$N_{\min} = 5\sqrt{X}, \tag{1}$$

where X is the number of training samples. Initialize the weights of each node (usually using random initialization). Randomly select a sample to input into the grid. Selecting winning neurons through competition:

$$\text{winner} = \arg\max_{\mathbf{w}_i} \|\mathbf{x} - \mathbf{w}_i\| . \tag{2}$$

The weight vectors serve as a reconstruction of the input data within the SOM. Learning-rate and neighborhood schedules:

$$\alpha(t) = \alpha(0) \cdot e^{t/K}, \tag{3}$$

$$j(t) = c + j(t-1) \cdot \big(1 - 0.001\, t\big), \tag{4}$$

where K is the iteration number, $j(t)$ is the number of domain scopes (decreasing over time), and $\alpha(t)$ is the learning rate function.

The blue spots represent the distribution of training data, while the small white spots are the current training data extracted from this distribution. Firstly, the SOM node is arbitrarily located in the data space. We choose the node closest to the training data as the winning node (highlighted in yellow). It is moved towards the training data, including (within a smaller range) adjacent nodes on its grid. After multiple iterations, the grid tends to approach the data distribution.

Randomize the node weight vector in a map, randomly select the input vector, traverse each node in the map, and calculate the similarity with the mapped nodes (usually using Euclidean distance); select the node with the smallest distance as the winning node BMU (best matching unit), and update the weight vector of nodes near BMU by bringing them

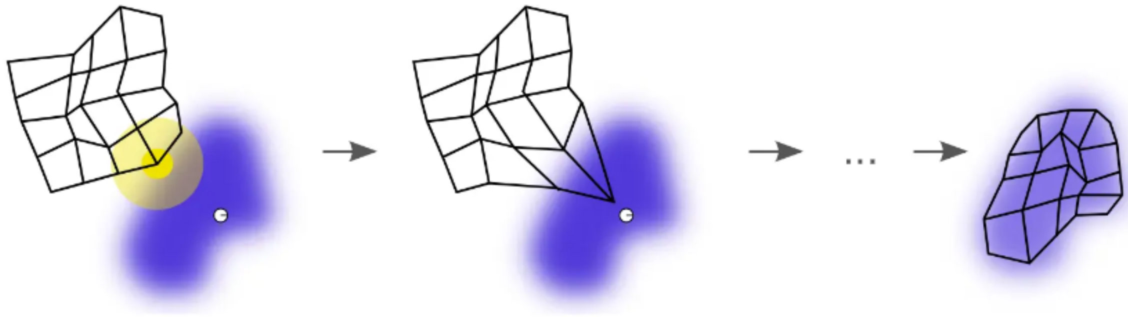

Figure 1. Self-Organizing Mapping Training.

closer to the input vector. The weight update formula for the output node with the current weight vector is:

$$\mathbf{W}_v(s+1) = \mathbf{W}_v(s) + \Theta(u, v, \ldots, s) \cdot \alpha(s) \cdot \big(\mathbf{D}(t) - \mathbf{W}_v(s)\big), \tag{5}$$

where s is the number of iterations, $\mathbf{D}(t)$ is the current input vector, u is the winning node, and $\Theta(\cdot)$ is a proximity function that gives the distance between u and v at step s, used to determine the strength of the influence of the winning node on its neighboring nodes. $\alpha(s)$ is a monotonically decreasing learning rate.

One indicator for measuring the quality of SOM is quantification (quantization) error, which is the squared norm between the input sample and the corresponding winner neuron:

$$E_q = \sum_{i=1}^{n} \left\|\mathbf{x}_i - \mathbf{w}_j\right\|^2 . \tag{6}$$

The neighborhood function is used to determine the strength of the influence of the winning node on its neighboring nodes, that is, the update amplitude of each node in the winning neighborhood. The most common choice is the Gaussian function, which can characterize the relationship between strength and distance within the superior neighborhood.

(2) Adaptive Learning Rates. Adaptive learning rates influence the adjustment of weight vectors during training. By adapting learning rates, the SOM can selectively update weight vectors based on the significance of input data for a particular neuron. The AdaGrad algorithm independently uses the learning rates of all model parameters, scaling each parameter inversely to the square root of the sum of the historical squared gradients. Cumulative squared gradient:

$$\mathbf{r} = \mathbf{r} + \mathbf{g} \odot \mathbf{g}, \tag{7}$$

calculate updates:

$$\Delta\boldsymbol{\theta} = \frac{\varepsilon}{\delta + \sqrt{\mathbf{r}}} \odot \mathbf{g}, \tag{8}$$

apply update:

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}, \tag{9}$$

where ε is the global learning rate, $\boldsymbol{\theta}$ is the parameter vector, and $\delta = 10^{-7}$.

(3) RMSProp (with momentum variants). The RMSProp algorithm modifies AdaGrad to achieve better performance under non-convex settings, changing gradient accumulation to an exponentially weighted moving average. Assume global learning rate ε, attenuation rate ρ, initial parameters $\boldsymbol{\theta}$, and small constant δ (usually 10^{-6} for numerical stability).

For a minibatch $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\}$ with targets $\{y^{(i)}\}$, compute the gradient:

$$\mathbf{g} = \frac{1}{m} \nabla_{\boldsymbol{\theta}} \sum_i \mathcal{L}\big(f(\mathbf{x}^{(i)}; \boldsymbol{\theta}), y^{(i)}\big). \tag{10}$$

Accumulate squared gradients:

$$\mathbf{r} = \rho\, \mathbf{r} + (1-\rho)\, \mathbf{g} \odot \mathbf{g}. \tag{11}$$

Parameter update:

$$\Delta\boldsymbol{\theta} = -\frac{\varepsilon}{\sqrt{\delta + \gamma}} \odot \mathbf{g} \left(\frac{1}{\sqrt{\delta + \gamma}} \right), \tag{12}$$

and apply

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}. \tag{13}$$

With momentum coefficient α and velocity $\mathbf{v}$:

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \alpha\, \mathbf{v}, \tag{14}$$

$$\mathbf{g} = \frac{1}{m} \nabla_{\hat{\boldsymbol{\theta}}} \sum_i \mathcal{L}\big(f(\mathbf{x}^{(i)}; \hat{\boldsymbol{\theta}}), y^{(i)}\big), \tag{15}$$

$$\mathbf{r} = \rho\, \mathbf{r} + (1-\rho)\, \mathbf{g} \odot \mathbf{g}, \tag{16}$$

$$\mathbf{v} = \alpha\, \mathbf{v} - \frac{\varepsilon}{\sqrt{\mathbf{r}}} \odot \mathbf{g} \left(\frac{1}{\sqrt{\mathbf{r}}} \right), \tag{17}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \mathbf{v}. \tag{18}$$

(4) Adam. With step size ε (recommended default 0.001), exponential decay rates of moment estimations ρ_1 and ρ_2 ($\in [0,1]$, defaults 0.9 and 0.999), and stability constant δ (default 10^{-8}), initialize first- and second-order moments $\mathbf{s} = \mathbf{0}$, $\mathbf{r} = \mathbf{0}$ and time step $t = 0$. For each minibatch:

$$\mathbf{g} = \frac{1}{m} \nabla_{\boldsymbol{\theta}} \sum_i \mathcal{L}\big(f(\mathbf{x}^{(i)}; \boldsymbol{\theta}), y^{(i)}\big), \tag{19}$$

$$t = t + 1, \tag{20}$$

$$\mathbf{s} = \rho_1\, \mathbf{s} + (1-\rho_1)\, \mathbf{g}, \tag{21}$$

and finally apply the parameter update (with bias correction implied):

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}. \tag{22}$$

Neighborhood functions. Neighborhood functions determine the influence of neighboring neurons during the training process. Advanced neighborhood functions allow for a more nuanced adjustment of weights, facilitating a finer reconstruction of input data.

2.2. **Common structures.** When referring to common structures, it usually involves identifying patterns or clusters within the SOM. In the context of product design structure optimization:

1. **Clustered Design Parameters:** Utilize the trained SOM to identify clusters or groups of similar design parameters. Neurons that are spatially close within the SOM tend to represent design parameters with similar characteristics.
2. **Visualization of Clusters:** Visualization techniques, such as heatmaps or 2D/3D projections, can be employed to represent clustered design parameters. These visualizations offer insights into common structures and relationships among design parameters.

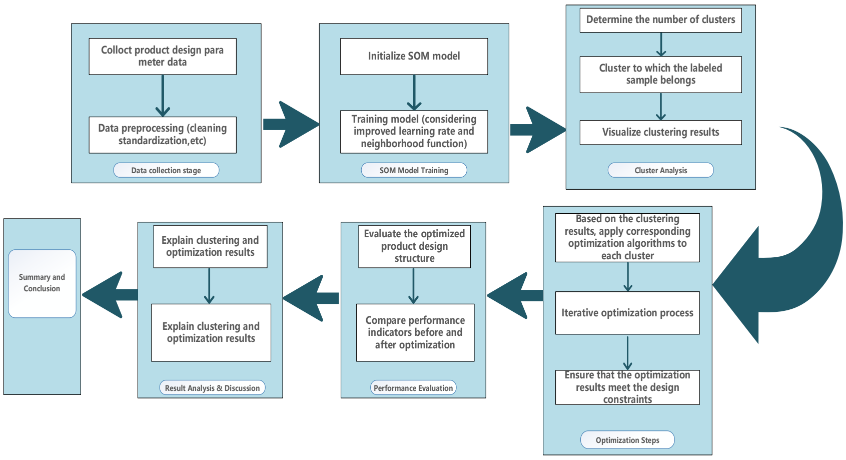

Figure 2. Algorithm framework architecture.

3. **Cluster-Specific Characteristics:** Analyze each cluster to identify common characteristics or patterns among design parameters within that cluster. Common structures may include specific combinations of design variables that lead to similar performance outcomes, as shown in Fig. 2.

Optimization within Clusters. Once common structures are identified, optimization algorithms can be applied specifically within each cluster. This approach allows for targeted optimization based on shared characteristics within the clusters. Optimization processes need to consider common structures while adhering to manufacturing constraints, cost limitations, and other relevant factors. Constraint-handling strategies ensure that optimized designs are feasible and practical.

3. **An enhanced SOM algorithm based on Adam with clustering and optimization techniques. Step 1. Data Collection and Preprocessing.** *Design Parameters:* Gather relevant design parameters, including geometric features, material properties, and other relevant variables. *Performance Metrics:* Define performance metrics such as efficiency, durability, or other criteria that reflect the objectives of the product design.

Step 2. Enhanced SOM Algorithm based on Adam. *Neuron Initialization:* Initialize the SOM with an appropriate topology and randomly distribute neurons in the SOM lattice. *Adaptive Learning Rates:* Implement adaptive learning rates to enhance the adaptability of the SOM to varying data distributions and non-linear relationships. *Advanced Neighborhood Functions:* Employ advanced neighborhood functions to capture complex relationships among design parameters.

Step 3. Training the Enhanced SOM based on Adam. *Input Vector Normalization:* Normalize input vectors to ensure uniformity in scaling and prevent dominance by certain parameters. *Iterative Training:* Iteratively present design parameter data to

Table 1. Configuration of DGA dataset

Product name	Original-SOM clustering	Improved-SOM clustering	Performance score	Production
Product A	Category 1	Category 1	85.3	30
Product B	Category 2	Category 1	90.8	25
Product C	Category 3	Category 2	87.5	28
Product D	Category 4	Category 3	89.6	26
Product E	Category 5	Category 4	92.3	24

the SOM, adjusting weights based on the adaptive learning rates and neighborhood functions. *Convergence Criteria:* Establish convergence criteria to determine when the SOM has sufficiently adapted to the data.

Step 4. Clustering Analysis. *Cluster Identification:* Utilize the trained SOM to identify clusters of design parameters within the SOM lattice. *Visualization:* Visualize the clustered design parameters to gain insights into the inherent structures and relationships.

4. Experimental results and analysis.

4.1. Experimental environment and experimental dataset.

Programming Language and Libraries. The enhanced Self-Organizing Map (SOM) algorithm and optimization routines are implemented in Python, leveraging machine learning and data-analysis libraries (e.g., `scikit-learn`).
Computational Resources. Experiments are conducted on a computing platform with sufficient CPU, memory, and optional GPU capabilities. Parallel processing is considered for efficient training and optimization, especially for large datasets.
Software and Versioning. Software versions are fixed and documented to ensure reproducibility.
Experiment Reproducibility. Clear instructions and documented code enable reproduction. Code and data are shared to encourage collaboration and validation.
Experimental Dataset. Table 1 summarizes the configuration of a representative DGA dataset comprising five products.
In this example, the original SOM and the proposed method yield distinct cluster assignments. The proposed method discovers stronger product similarity (e.g., grouping A and B), while improving performance and reducing cost (notably for Product E).
Source of Data. Real-world datasets from relevant industries are used, representative of actual complexities and variations.
Multidimensional Design Parameters. Design parameters include geometric features, material properties, and other variables that influence performance.
Performance Metrics. Metrics align with design objectives, e.g., efficiency, durability, and cost-effectiveness.
Data Preprocessing. Data are cleaned and normalized for SOM training and subsequent optimization.
Dataset Size and Splits. Multiple dataset sizes are considered to assess scalability. Training/testing splits are used to evaluate generalization, with testing sets representing unseen data.
Privacy and Ethics. Ethical and privacy guidelines are followed; proprietary data are anonymized and secured.
Dataset Diversity. Datasets span varied product types and industries to demonstrate adaptability.

Table 2. Comparison of results with other algorithms

Algorithm	Stability	Sensitivity	Time (h)	Best score	Avg. score	Accuracy (%
Original SOM	commonly	high	2.5	88.2	85.3	78.9
K-means	commonly	low	1.8	87.9	85.7	86.4
DBSCAN	preferably	low	3.2	88.5	86.1	89.3
Spectral Clustering	preferably	low	2.1	88.6	86.3	90.2
Improved-SOM Clustering	preferably	low	1.5	90.8	87.2	94.5

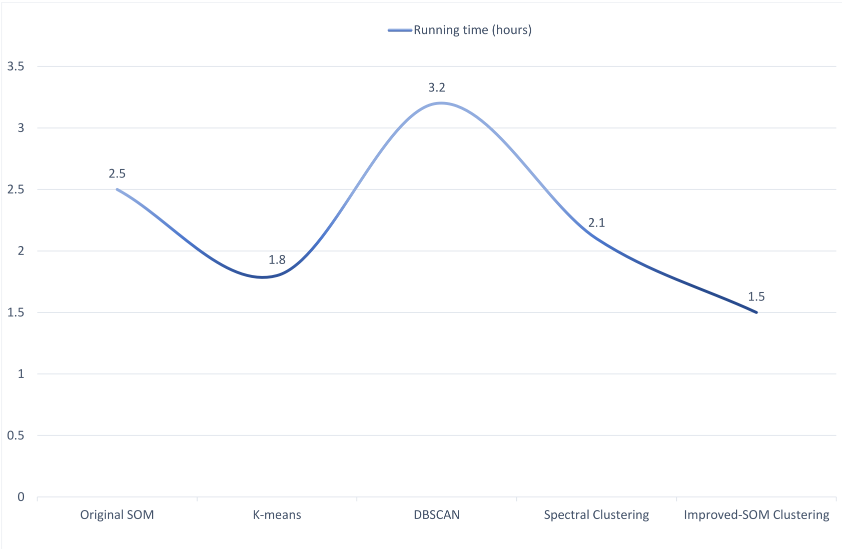

Figure 3. Comparison of running time with other algorithms.

Table 3. Performance of different products

Product Name	Material type	Weight (g)	Cost (yuan)	Performance rating	Optimization e
Product A	Plastic	120	1000	70	5%
Product B	Metal	250	2000	85	10%
Product C	Composite Material	380	3000	92	8%
Product D	Ceramic	500	4000	95	15%

4.2. **Experimental results.** To benchmark clustering quality and downstream design impact, we compare Original SOM, K-means, DBSCAN, Spectral Clustering, and the proposed Improved-SOM Clustering.

As shown in Figs. 3–4 and Table 2, the proposed method exhibits stronger clustering stability, lower sensitivity to initialization, and reduced runtime, while attaining the highest best/average performance scores and accuracy. The improved SOM effectively discovers similarity structures among designs, scales to larger datasets, and handles complex, non-linear data manifolds.

Table 3 reports exemplar per-product outcomes, highlighting consistent optimization gains from cluster-aware refinement.

Figure 4. Comparison of clustering results and outcome metrics across algorithms.

5. **Conclusions.** The research on product design structure optimization through an enhanced SOM algorithm has yielded several insights. Adaptive learning rates and advanced neighborhood functions improve adaptability to high-dimensional, non-linear design spaces, enabling the SOM to capture intricate parameter relationships. Clustering analysis reveals common structures, and targeted optimization within clusters leads to improved performance while respecting manufacturing and cost constraints. Empirical evaluation on representative datasets demonstrates practical applicability and robustness across scenarios.

Limitations include sensitivity to extreme design complexities, the need for parameter fine-tuning, and scaling to extremely large datasets. Future work will pursue richer neighborhood schedules, dynamic map growth, and tighter integration with constraint-aware optimizers.

Acknowledgment. This work is partially supported by the Key projects of Xuzhou Vocational College of Industrial Technology (No. XGY2023ZXZD03).

REFERENCES

[1] J. Liao, A. Jantan, Y. Ruan, and C. Zhou, "Multi-Behavior RFM Model Based on Improved SOM Neural Network Algorithm for Customer Segmentation," *IEEE Access*, vol. 10, pp. 122501–122512, 2022.
[2] Z. Xia, J. Mao, and Y. He, "Rockburst intensity prediction in underground buildings based on improved spectral clustering algorithm," *Frontiers in Earth Science*, vol. 10, p. 4562, 2022.
[3] Y. Wang, Y. Liang, H. Sun, and Y. Ma, "RETRACTED ARTICLE: Intrusion detection and performance simulation based on improved sequential pattern mining algorithm," *Cluster Computing*, vol. 23, no. 3, pp. 1927–1936, 2020. (Retracted)
[4] C. Yang, S. Su, X. Ju, and J. Song, "A Mobile Sensors Dispatch Scheme Based on Improved SOM Algorithm for Coverage Hole Healing," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 21080–21089, 2021.

[5] W. Wang, Y. Yan, R. Zhang, Z. Wang, Y. Fan, and C. Yang, "Multi-Attribute Fusion Algorithm Based on Improved Evidence Theory and Clustering," *Sensors*, vol. 19, no. 19, p. 4146, 2019.

[6] K. Rong, "Analysis of the Comprehensive Evaluation Model of Enterprise Technological Innovation Ability Based on Improved Genetic Algorithm," *Journal of Mathematics*, vol. 2022, p. 40, 2022.

[7] D. Xiao, H. Li, and X. Sun, "Coal Classification Method Based on Improved Local Receptive Field-Based Extreme Learning Machine Algorithm and Visible–Infrared Spectroscopy," *ACS Omega*, vol. 5, no. 40, pp. 25772–25783, 2020.

[8] S.-S. Yu, S.-W. Chu, C.-M. Wang, Y.-K. Chan, and T.-C. Chang, "Two improved k-means algorithms," *Applied Soft Computing*, vol. 68, pp. 747–755, 2018.

[9] G. Ding, X. Pei, Y. Yang, and B. Huang, "Segmentation of the Fabric Pattern Based on Improved Fruit Fly Optimization Algorithm," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–7, 2020.

[10] L. Zhang, F. Qiao, J. Wang, and X. Zhai, "Equipment Health Assessment Based on Improved Incremental Support Vector Data Description," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 5, pp. 3205–3216, 2021.

[11] Y. Huo, Y. Cao, Z. Wang, Y. Yan, Z. Ge, and Y. Yang, "Traffic Anomaly Detection Method Based on Improved GRU and EFMS-Kmeans Clustering," *Computer Modeling in Engineering & Sciences*, vol. 126, no. 3, pp. 1053–1091, 2021.

[12] C. Yan, M. Li, and W. Liu, "Prediction of bank telephone marketing results based on improved whale algorithms optimizing S_Kohonen network," *Applied Soft Computing*, vol. 92, p. 106259, 2020.

[13] C. Lu, J. Yang, B. J. Leira, Q. Chen, and S. Wang, "Three-Dimensional Path Planning of Deep-Sea Mining Vehicle Based on Improved Particle Swarm Optimization," *Journal of Marine Science and Engineering*, vol. 11, no. 9, p. 1797, 2023.

[14] H. Guo, M. Li, X. Zhang, Q. Liu, and X. Gao, "Research on indoor wireless positioning precision optimization based on UWB," *Journal of Web Engineering*, vol. 19, no. 7–8, pp. 1017–1048, 2020.

[15] W. Ma, Y. Zhang, J. Guo, and K. Li, "Unbalanced abnormal traffic detection based on improved Res-BIGRU and integrated dynamic ELM optimization," *Computer Communications*, vol. 179, pp. 112–130, 2021.

[16] W. Liu, R. Tan, G. Cao, F. Yu, and H. Li, "Creative design through knowledge clustering and case-based reasoning," *Engineering with Computers*, vol. 36, pp. 527–541, 2020.

[17] J. Wang, Y. Liu, and H. Feng, "IFACNN: efficient DDoS attack detection based on improved firefly algorithm to optimize convolutional neural networks," *Mathematical Biosciences and Engineering*, vol. 19, no. 2, pp. 1280–1303, 2021.

[18] Z. Zhao, X. Yang, Y. Zhou, Q. Sun, Z. Ge, and D. Liu, "Real-time detection of particleboard surface defects based on improved YOLOv5 target detection," *Scientific Reports*, vol. 11, p. 15624, 2021.

[19] E. E. Seghers, L. A. Briceno-Mena, and J. A. Romagnoli, "Unsupervised learning: Local and global structure preservation in industrial data," *Computers & Chemical Engineering*, vol. 178, p. 108378, 2023.

[20] L. Ma, L. Zhao, L. Cao, D. Li, G. Chen, and Y. Han, "Inversion of Soil Organic Matter Content Based on Improved Convolutional Neural Network," *Sensors*, vol. 22, no. 20, p. 7777, 2022.

[21] C. Wang, "Efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach," *Information Processing & Management*, vol. 59, no. 6, p. 103085, 2022.

[22] Q. Ran, Z. Liu, X. Sun, X. Sun, B. Zhang, Q. Guo, and J. Wang, "Anomaly Detection for Hyperspectral Images Based on Improved Low-Rank and Sparse Representation and Joint Gaussian Mixture Distribution," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 6339–6352, 2021.

[23] N. Jiang and T. Liu, "An Improved Speech Segmentation and Clustering Algorithm Based on SOM and K-Means," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–19, 2020.

[24] F. He, J. Yang, Y. Zhang, W. Yu, X. Xiao, and J. Xia, "Does partition matter? A new approach to modeling land use change," *Computers, Environment and Urban Systems*, vol. 106, p. 102041, 2023.

[25] Y. Cai, X. Wang, and L. Xiong, "[Retracted] Difference Analysis of Regional Economic Development Based on the SOM Neural Network with the Hybrid Genetic Algorithm," *Computational Intelligence and Neuroscience*, vol. 2021, p. 562, 2021. (Retracted)

[26] A. Yan, W. Wang, Y. Ren, and H. Geng, "A Clustering Algorithm for Multi-Modal Heterogeneous Big Data With Abnormal Data," *Frontiers in Neurorobotics*, vol. 15, p. 2141, 2021.

[27] J. Liu and K. Li, "Research on an Improved SOM Model for Damage Identification of Concrete Structures," *Applied Sciences*, vol. 12, no. 9, p. 4152, 2022.

[28] H. Liu and H. Zhao, "Upgrading models, evolutionary mechanisms and vertical cases of service-oriented manufacturing in SVC leading enterprises: Product-development and service-innovation for industry 4.0," *Humanities and Social Sciences Communications*, vol. 9, p. 1479, 2022.

[29] H. Tong, W. Lu, W. Xing, and S. Wang, "ARRAY: Adaptive triple feature-weighted transfer Naive Bayes for cross-project defect prediction," *Journal of Systems and Software*, vol. 202, p. 111721, 2023.