

# Efficiency of optimization algorithms in the problem of graphic objects placement\*

Bohdana Havrysh<sup>1,\*</sup>, Dmytro Palamarchuk<sup>1,\*</sup>, Oleksandr Tymchenko<sup>1,\*</sup>, Bohdan Kovalskiy<sup>1,†</sup> and Orest Khamula<sup>1,‡</sup>

<sup>1</sup> Lviv Polytechnic National University, Stepana Bandery Street, 12, Lviv, 79000, Ukraine

## Abstract

The article considers a comprehensive comparison of optimization algorithms for solving the problem of placing vector graphic objects on a plane. Algorithms are described in the form of block diagrams, which allows an understanding of their structure and operation principles. These diagrams provide a step-by-step visual representation of the processes involved, making it easier to follow the logical process and identify the critical components of each algorithm. Software was developed to evaluate the effectiveness of these methods. This program uses simplified input data in the form of rectangular objects that are placed on planes of different sizes. A set of rectangular objects and different plane dimensions makes it possible to perform various analyses in different scenarios, guaranteeing the reliability of the results. Research results are carefully presented in graphs and tables that clearly illustrate the effectiveness of each method. Furthermore, the article thoroughly discusses the results and conclusions obtained, indicating the high potential of optimization search methods in solving the problem of placing graphic objects. It highlights the practical implications of these findings, suggesting that these methods can significantly improve efficiency and resource utilization in various applications, such as manufacturing, printing, and layout design. The results of this study can guide future development and innovation in optimization methods, contributing to advances in fields that require accurate and efficient feature placement strategies.

## Keywords

optimization, comparative analysis, vector graphic objects, genetic algorithm, simulation modeling, annealing simulation method

## 1. Introduction

Placing vector objects on a plane is a common challenge in various industries, such as cutting shapes from paper, metal, wood, and other substrates [1, 21]. The optimal arrangement becomes essential when cutting many pieces, as it helps conserve material and handle non-standard plane shapes efficiently. Optimizing the placement of vector objects on a plane is a complex problem due to the vast number of possible configurations and the need to consider multiple parameters for each object. For example, the rotation of objects can significantly influence the overall efficiency of the placement. Furthermore, maintaining sufficient spacing between objects is vital to prevent material damage or other production defects.

This optimization challenge belongs to the category of NP-hard problems, making exact methods impractical for large datasets. Therefore, it is more suitable to use heuristic optimization methods that can provide acceptable solutions within a reasonable time frame. This paper explores three optimization methods: the genetic algorithm, simulated annealing, and simulation modeling. A software application was developed to evaluate the effectiveness of these algorithms by processing

---

*AdvAIT-2024: 1st International Workshop on Advanced Applied Information Technologies, December 5, 2024, Khmelnytskyi, Ukraine - Zilina, Slovakia*

\* Corresponding author.

† These authors contributed equally.

✉ dana.havrysh@gmail.com (B. Havrysh); dmytro.palamarchuk@gmail.com (D. Palamarchuk);

alextymchenko53@gmail.com (O. Tymchenko); bkovalskiy@ukr.net (B. Kovalskiy); khamula@gmail.com (O. Khamula)

ORCID 0000-0003-3213-9747 (B. Havrysh), 0009-0000-3934-5899 (D. Palamarchuk); 0000-0001-6315-9375 (O. Tymchenko);

0000-0001-9088-1144 (B. Kovalskiy); 0000-0003-0926-9156 (O. Khamula)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

simplified input data, which consisted of rectangular objects placed on planes of various sizes. The findings of this research may contribute to the creation of more efficient algorithms for optimization tasks across various industrial and applied fields, enabling resource savings and improving productivity.

## **2. Analysis of recent researches and publications**

Due to its importance and complexity, the problem of finding optimal solutions is actively studied in various scientific and applied fields. A variety of methods developed to tackle this problem, with the most widely used being genetic algorithms, simulated annealing, and simulation modeling. Each of these methods offers unique approaches to finding optimal solutions.

*Simulated Annealing* - is an optimization algorithm miming the physical process of annealing metals [2, 13-15]. In this method, the system is "heated" to explore various possibilities and then "cooled" to avoid getting trapped in local minima. This algorithm is effective for optimizing complex spaces with many local optima. It offers several key advantages, including simplicity of implementation and the ability to find optimal solutions for various problems. However, it may require significant time and parameter tuning. Due to its flexibility and ability to find optimal solutions in challenging environments, simulated annealing is widely applied to optimization tasks such as the traveling salesperson problem, scheduling, and task allocation.

*Simulation modeling* is a powerful tool in applied systems analysis that allows the exploration of complex systems and processes, mainly when decision-making is linked to uncertainty [3, 18-20]. Compared to other methods, it enables consideration of many alternatives, improves the quality of management decisions, and forecasts their consequences. However, using simulation modeling in practical management is relatively common due to the complexity of the mathematical framework and the processing of large datasets. The method is based on replicating the functioning process of a system, including its interaction with the external environment, and its model can be represented as functionally implemented blocks, either in software or hardware. Simulation modeling allows include of a wide range of factors in decision-making. It is one of the most powerful tools for analyzing complex systems and processes, particularly under conditions of subjectivity and uncertainty, where reliable analytical tools are required.

*Genetic algorithm* stands apart from traditional optimization methods and offers several key advantages [4, 15-17]. It operates on encoded values and searches within a population, enabling efficient solution space exploration. This method does not require objective function derivatives, employs probabilistic selection rules to avoid getting stuck in local optima, and features inherent parallelism to explore multiple search directions simultaneously. It excels in complex search domains, can manage numerous parameters concurrently, and requires no additional problem-specific information while delivering solutions quickly. However, a significant limitation is the inability to create a universal code for describing functions and optimization criteria due to varying initial conditions. Genetic algorithms are widely applied across scientific and technical fields, including neural network design, production management, economics, medicine, mathematics, etc [5, 6, 22].

## **3. Objective of the work**

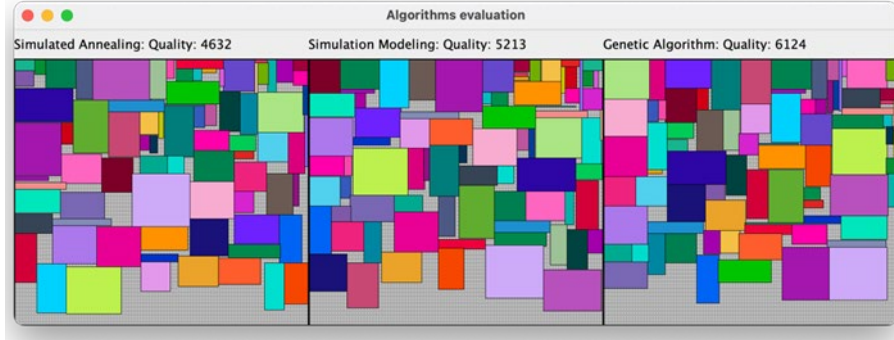
Analyze and compare the effectiveness of simulated annealing, simulation modeling, and the genetic algorithm for efficiently placing vector graphic objects on a plane.

## **4. Presentation of the main research material**

A software application was developed to demonstrate the performance of the algorithms and compare their efficiency in optimizing the placement of vector objects on a given plane. This application was implemented using the Java Swing library [5, 7, 23] and written in Kotlin. The

application's input data includes the board's size and the number of objects to be placed on it. To simplify the problem, only rectangular shapes were used, ranging in size from 1 to 40 cells in height and width, with randomly generated dimensions. The goal of the application is to place the shapes on the plane while occupying the least available space. The shapes are arranged row by row, starting from the top left corner of the plane. The application evaluates the quality of the solution based on the number of empty cells remaining from the bottom-right corner.

In the software application (Figure 1), three planes are displayed simultaneously, each using identical input data but applying different algorithms: simulated annealing, simulation modeling, and the genetic algorithm. The quality of the resulting solution is displayed above each plane. Naturally, the higher the quality score, the better the result.



**Figure 1:** Software application.

#### 4.1. Simulated annealing method

The simulated annealing method has several variations that differ in the temperature change laws [6, 10-12]. Each variation has advantages and disadvantages, such as speed, the guarantee of finding the global minimum, and implementation complexity. A modification of the algorithm called "Very Fast Annealing" was selected for this task. To better understand how the algorithm works in the problem of optimal object placement on a plane, it can be represented as the following sequence of steps:

1. Two values are defined: the initial temperature  $T_0$  and the final temperature  $T_{end}$ .
2. A valid initial solution  $x_0$  is generated, its quality is measured, and the resulting evaluation is saved as  $r_0$ .
3. The initial temperature is assigned to the current temperature,  $T = T_0$ .
4. A new iteration is initiated. In each iteration, two shapes selected by a random number generator are swapped. New shape indices are recalculated if the chosen shapes cannot be placed in the new positions. The swap is performed once the new indices are determined, resulting in a new solution  $x$ .
5. The new solution is evaluated using the function  $r=f(x)$ . If  $r < r_0$ , the solution has improved, and the new solution is stored as the current one  $x_0=x$ ,  $r_0=r$ . Otherwise, if  $r > r_0$ , the new solution is accepted as the current one only with the probability:

$$h(\Delta E, T) = \frac{1}{1 + \exp(\Delta E / T)}, \quad (1)$$

$\Delta E = r_0 - r$ . Thus, the probability of accepting a worse solution as the current one decreases as the temperature lowers and the difference between the current energy  $r$  and the optimal energy  $r_0$ .

6. The new temperature is calculated as follows:

$$T_i(k) = T_{(i,0)} \exp(c_i k^{1/D}), c_i > 0, \quad (2)$$

where  $D$  – is the total number of iterations,  $k$  – is the current iteration number,  $c$  – is a variable typically used in the general pathfinding problem in a coordinate system to determine the specific temperature at each point. For the optimization problem, we simplify and keep it as a constant  $c=1$ . The temperature will decrease according to an exponential law.

7. If  $T > T_{end}$ , proceed to step 4 and continue the search, otherwise, terminate the algorithm.

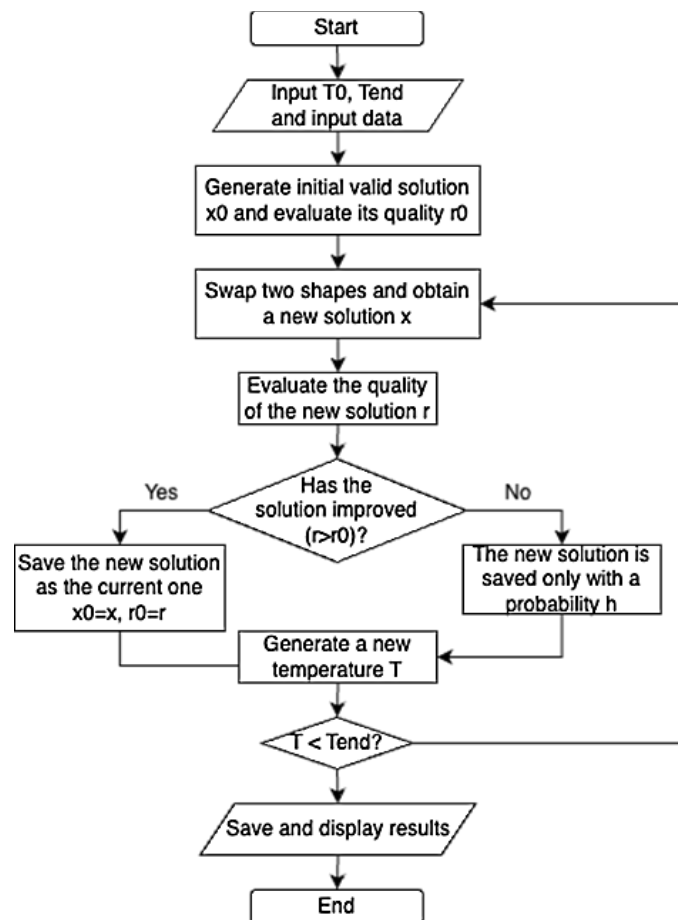
The flowchart of the algorithm is shown in Figure 2.

## 4.2. Simulation modeling

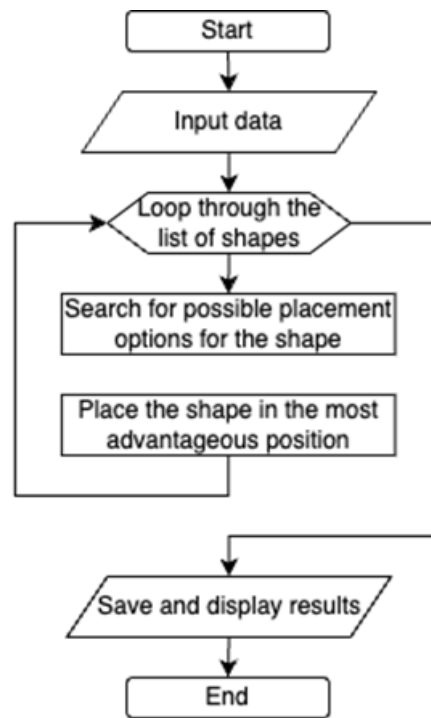
Simulation modeling is an approach to research in which the natural system is replaced by a model that accurately describes its functioning [8, 9, 24]. In the context of optimizing the placement of shapes on a plane, this algorithm can be broken down into the following steps:

1. Select a shape that still needs to be placed.
2. Identify all potential placement options on the plane for the selected shape that satisfy its strict constraints and compile them into the *list*.
3. Evaluate the quality of each position and place the shape in the most optimal option.

The flowchart in Figure 3 illustrates the working principle of the algorithm.



**Figure 2:** Flowchart of the simulated annealing method.



**Figure 3:** Flowchart of the simulation modeling method.

### 4.3. Genetic algorithm

The genetic algorithm is based on natural selection, where better-adapted species are more likely to survive [21, 25]. The application of the genetic algorithm for optimizing the placement of shapes on a plane can be described through the following steps:

1. Input the required number of iterations  $N$ , which has been determined experimentally. Use the variable  $count=0$  as a counter.
2. Retrieve the list of shapes that have not yet been placed. Based on this, generate an initial valid solution  $x_0$ . Evaluate this solution using the method  $f(x)$  and store the result as  $r_0$ .
3. Randomly change the order of shapes in the list and generate a new valid solution  $x$  based on it, then evaluate the solution as  $r=f(x)$ .
4. If  $r < r_0$ , save the new solution as the current one  $x_0=x$ ,  $r_0=r$ .
5. Increment the counter  $count++$ . If count equals  $N$ , terminate the search.

The sequence of steps for executing the algorithm is shown in the flowchart in Figure 4.

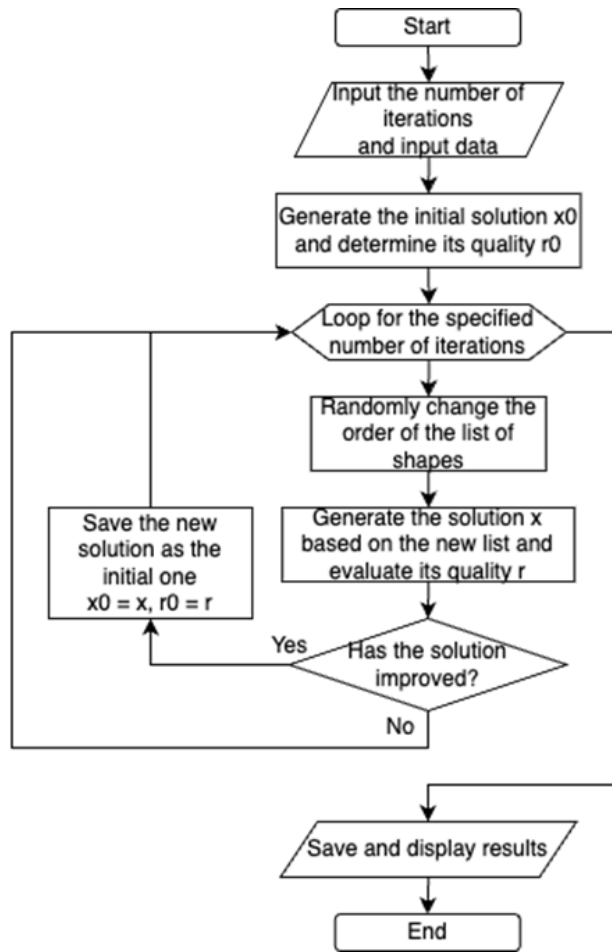
## 5. Research execution

Three different types of input data were used across 20 experiments to compare and evaluate the performance of each algorithm:

1. a 200x200 grid with 80 shapes,
2. a 300x300 grid with 170 shapes,
3. a 400x400 grid with 300 shapes.

The grid size, number of shapes, and number of iterations were selected experimentally to ensure the algorithms could operate with different grid sizes. In contrast, the number of shapes would not exceed the grid's capacity. Additionally, the solution search time was kept consistent across different

methods. The experiment results are presented in Table 1, which provides the quality scores of the outcomes for each algorithm.



**Figure 4:** Flowchart of the genetic algorithm operation.

**Table 1**

Quality assessments of the algorithm performance results

Grid size: 200x200			Grid size: 300x300			Grid size: 400x400		
Number of shapes: 80			Number of shapes: 170			Number of shapes: 300		
SA	SM	GA	SA	SM	GA	SA	SM	GA
5509	5433	6376	16168	16191	17440	21593	21208	23196
5652	5804	6426	16223	17286	17419	21593	21405	23239
5649	6272	6225	16255	17061	18054	21593	22523	23274
5652	5797	6785	16168	16892	17364	21593	22464	23067
5645	5727	6242	16168	16285	17327	21593	23253	23439
5652	5710	6877	16168	17391	17443	21601	22335	23297
5652	5389	6357	16168	17242	17566	21593	21525	23135
5652	5623	6429	16168	15603	17434	21593	21632	23558
5643	4843	6295	16168	16665	17657	21593	21067	23212
5652	5927	6275	16108	16503	17787	21564	21607	23470
5652	6335	6373	16255	16250	17603	21641	22228	23381
5652	5068	6529	16168	16310	17496	21593	21811	23587
5652	5166	6521	16168	16592	17856	21676	21667	23255
5652	5642	6357	16168	16852	17820	21593	20690	23512
5649	5432	6405	16177	17237	17449	21593	21648	22903

5640	5991	6509	16168	17158	17354	21593	22392	23058
5640	5978	6391	16168	16881	17766	21593	21702	23143
5652	5957	6611	16168	16856	17582	21593	22635	23521
5645	5762	6314	16223	16155	17734	21593	21343	23329
5640	5828	6313	13129	16457	17593	21593	21811	23380

Figures 5-7 display graphs to compare the algorithms' performance. The vertical axis represents the quality scores of the results, while the horizontal axis shows the iteration numbers of the experiments.

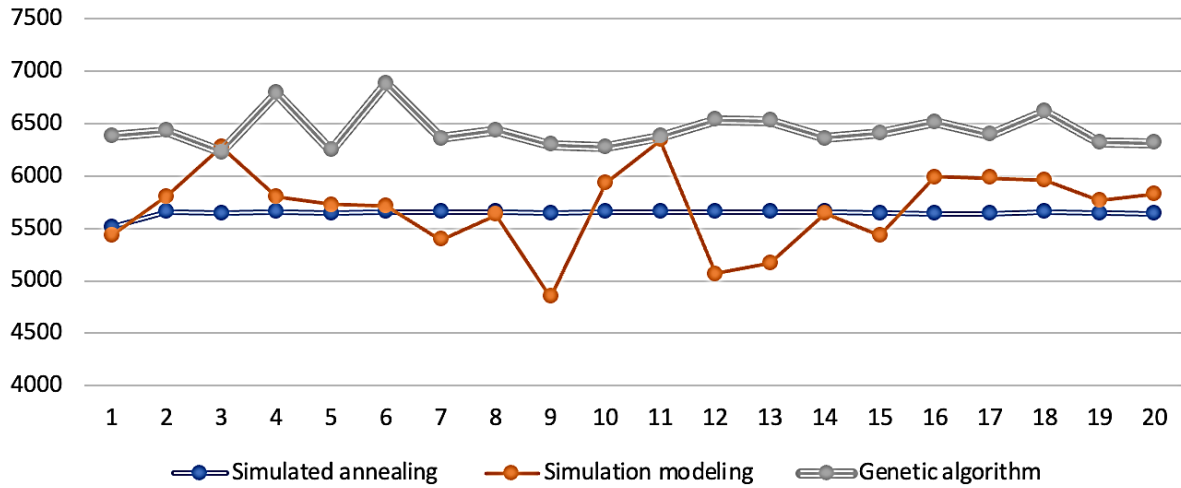


Figure 5: Comparison of results on a 200x200 grid.

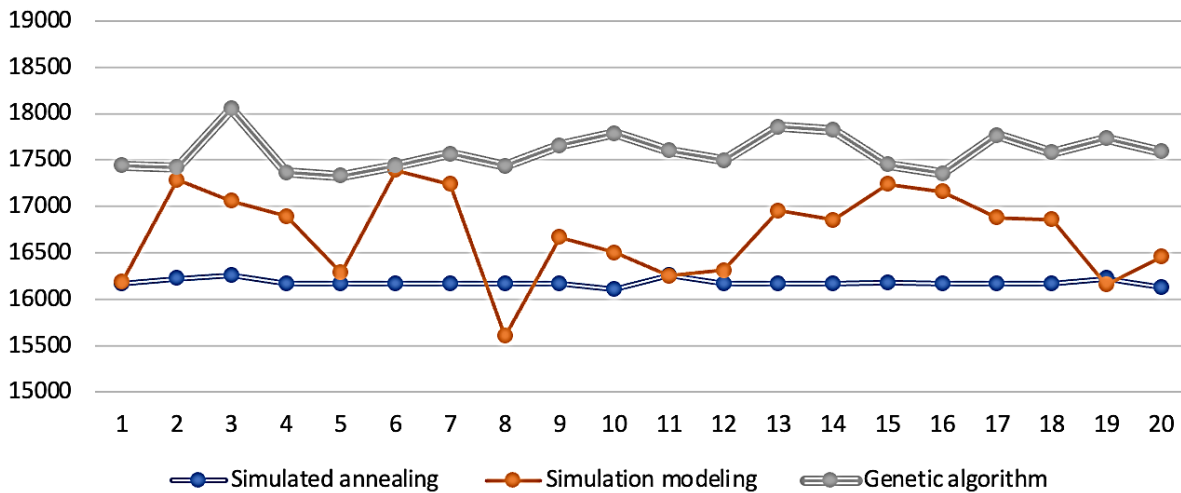


Figure 6: Comparison of results on a 300x300 grid.

The genetic algorithm consistently delivers the best results across all grid sizes from the presented graphs. The simulated annealing method shows lower efficiency, as it is more difficult to apply to this problem. It requires finding two shapes that can be swapped, which is only sometimes feasible. The simulation modeling method sometimes produces results close to those of the genetic algorithms but lacks sufficient stability.

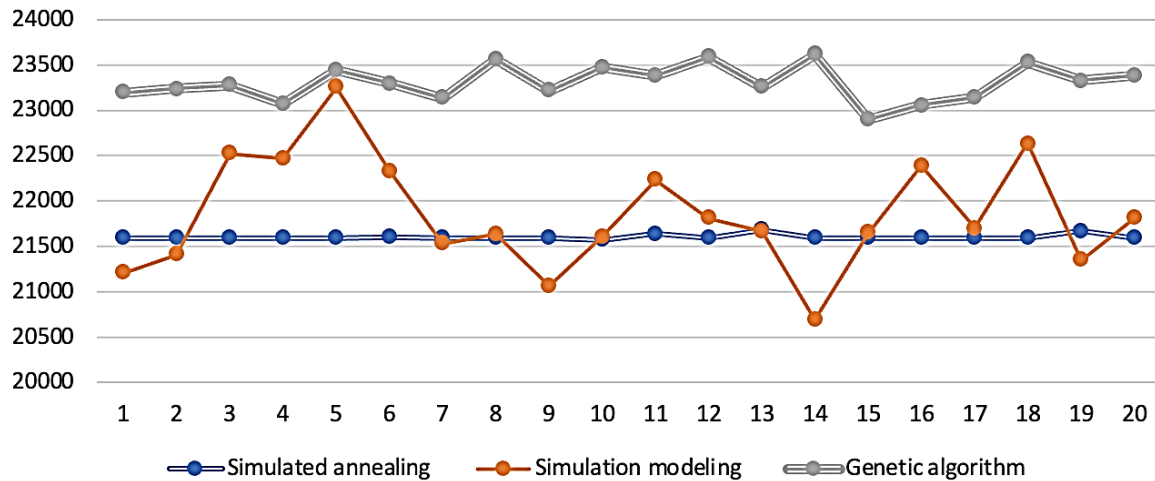


Figure 7: Comparison of results on a 400x400 grid.

## 6. Conclusions

This paper analyzes applying optimization methods to place vector graphic objects on a plane. To solve this problem, the paper proposes using heuristic optimization methods: genetic algorithms, simulated annealing, and simulation modeling. A software application was developed to compare the effectiveness of these algorithms, using simplified input data in the form of rectangular objects placed on grids of various sizes.

The genetic algorithm showed consistently high simulation results, which are 10-20% better than the method of simulating annealing on all planes. To use the latter, it is necessary to find the shapes that need to be swapped, which is not always possible. It is not advisable to use the simulation modeling method because it shows unstable results that are sometimes close to the genetic algorithm, but can be worse than the annealing simulation method.

The results of this research could contribute to developing more efficient algorithms for optimization tasks in various industrial and applied fields, leading to resource savings and increased productivity. The findings highlight the strong potential of genetic algorithms for optimization tasks involving the placement of graphic objects while also indicating the need for further research to improve these methods and enhance their efficiency under different conditions.

## Acknowledgements

The authors are appreciative of colleagues for their support and appropriate suggestions, which allowed to improve the materials of the article.

## Declaration on Generative AI

The authors have not employed any Generative AI tools.

## References

- [1] J. Jiang, "Research and Application of Object Recognition Algorithm in Architectural Engineering Drawing Based on Deep Learning," 2023 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), Dalian, China, 2023, pp. 121-124, doi: 10.1109/IPEC57296.2023.00029.
- [2] E. J. Bayro-Corrochano, G. Altamirano-Escobedo, A. Ortiz-Gonzalez, V. Farias-Moreno and N. Chel-Puc, "Computing in the Conformal Space Objects, Incidence Relations, and Geometric Constrains for Applications in AI, GIS, Graphics, Robotics, and Human-Machine Interaction," in IEEE Access, vol. 10, pp. 112742-112756, 2022, doi: 10.1109/ACCESS.2022.3216266.



- [3] S. Xu, R. Wang, H. Wang and R. Yang, "Plane Segmentation Based on the Optimal-Vector-Field in LiDAR Point Clouds," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3991-4007, 1 Nov. 2021, doi: 10.1109/TPAMI.2020.2994935.
- [4] Z. Fan, T. Chen, P. Wang and Z. Wang, "CADTransformer: Panoptic Symbol Spotting Transformer for CAD Drawings," 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, 2022, pp. 10976-10986, doi: 10.1109/CVPR52688.2022.01071.
- [5] O.Tymchenko, B.Havrysh, O.O. Tymchenko, O. Khamula, B. Kovalskyi, K. Havrysh. Person voice recognition methods (2020) *Proceedings of the 2020 IEEE 3rd International Conference on Data Stream Mining and Processing, DSMP 2020*, art. no. 9204023, pp. 287-290. ISBN: 978-172813214-3 doi: 10.1109/DSMP47368.2020.9204023.
- [6] T. Hehn, J. Kooij and D. Gavrilu, "Fast and Compact Image Segmentation Using Instance Stixels," in *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 1, pp. 45-56, March 2022, doi: 10.1109/TIV.2021.3067223.
- [7] D. Qiao and E. Sun, "Research on Parallel Machine Scheduling Based on Genetic Neighborhood Hybrid Algorithm," 2019 2nd International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2019, pp. 324-327, doi: 10.1109/ICISCAE48440.2019.221644.
- [8] L. Yichen, L. Bo, Z. Chenqian and M. Teng, "Intelligent Frequency Assignment Algorithm Based on Hybrid Genetic Algorithm," 2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL), Chongqing, China, 2020, pp. 461-467, doi: 10.1109/CVIDL51233.2020.00-50.
- [9] D. Chaudhary, A. K. Tailor, V. P. Sharma and S. Chaturvedi, "HyGADE: Hybrid of Genetic Algorithm and Differential Evolution Algorithm," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019, pp. 1-4, doi: 10.1109/ICCCNT45670.2019.8944822.
- [10] O. Tymchenko, B. Havrysh, O. Khamula, S. Lysenko, O.O. Tymchenko, K. Havrysh. Risks of loss of personal data in the process of sending and printing documents (Open Access). *CEUR Workshop Proceedings*, 2020, vol. 2805, pp. 373-384.
- [11] T. Wu, L. Wang, H. Huang, Z. Lai and Q. Ling, "A Multi-population Adaptive Genetic Algorithm for Test Paper Generation," 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 2021, pp. 5157-5162, doi: 10.1109/CCDC52312.2021.9602353.
- [12] M. A. FuRui, Al-Absi and H. J. Lee, "Introduce a Specific Process of Genetic Algorithm through an Example," 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea (South), 2019, pp. 422-425, doi: 10.1109/ICTC46691.2019.8939728.
- [13] W. Firgiawan, S. Cokrowibowo, A. Irianti and A. Gunawan, "Performance Comparison of Spider Monkey Optimization and Genetic Algorithm for Traveling Salesman Problem," 2021 3rd International Conference on Electronics Representation and Algorithm (ICERA), Yogyakarta, Indonesia, 2021, pp. 191-195, doi: 10.1109/ICERA53111.2021.9538795.
- [14] B. Durnyak, B. Havrysh, O. Tymchenko, M. Zelyanovsky, O. O. Tymchenko and O. Khamula, Intelligent System for Sensor Wireless Network Access: Modeling Methods of Network Construction, *IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)*, Lviv, Ukraine, 2018, pp. 93-97, doi: 10.1109/IDAACS-SWS.2018.8525792.
- [15] Y. LIU and W. LI, "An ATO Multi-objective Optimization Control Strategy Based on Genetic Algorithm," 2019 Chinese Control And Decision Conference (CCDC), Nanchang, China, 2019, pp. 1214-1218, doi: 10.1109/CCDC.2019.8832917.
- [16] Y. Shi, L. Qin, D. Zhao and Y. Xu, "Research on Indoor Robot Localization Method Based on Clustering Optimizes Genetic Algorithm," 2023 2nd International Conference on Artificial Intelligence and Intelligent Information Processing (AIIP), Hangzhou, China, 2023, pp. 153-158, doi: 10.1109/AIIP61647.2023.00035.

- [17] A. Raj, A. Kumar, V. Sharma, S. Rani, A. K. Shanu and T. Singh, "Applications of Genetic Algorithm with Integrated Machine Learning," 2023 3rd International Conference on Innovative Practices in Technology and Management (ICIPTM), Uttar Pradesh, India, 2023, pp. 1-6, doi: 10.1109/ICIPTM57143.2023.10118328.
- [18] W. Jia and Z. He, "Outlier data Mining of large Data Sets relying on fast decomposition simulated annealing algorithm," 2020 International Conference on Robots & Intelligent System (ICRIS), Sanya, China, 2020, pp. 679-682, doi: 10.1109/ICRIS52159.2020.00170.
- [19] B. Aylaj and S. Nouh, "Degeneration vs Classical of simulated annealing algorithm: performance analysis," 2022 5th International Conference on Advanced Communication Technologies and Networking (CommNet), Marrakech, Morocco, 2022, pp. 1-5, doi: 10.1109/CommNet56067.2022.9993814.
- [20] I. Zasornova, T. Hovorushchenko, O.Voichur. Study of software testing tools according to the testing levels. Computer Systems and Information Technologies. 2023.No 1. P. 38-46.
- [21] J. Zhao and H. Wang, "Test Paper Generation Based on Improved Genetic Simulated Annealing Algorithm," 2022 12th International Conference on Information Technology in Medicine and Education (ITME), Xiamen, China, 2022, pp. 384-388, doi: 10.1109/ITME56794.2022.00089.
- [22] A. Phu-Ang, "Solving The Travelling Salesman Problem by Using Hybrid Simulated Annealing with Tabu Technique," 2022 3rd International Conference on Big Data Analytics and Practices (IBDAP), Bangkok, Thailand, 2022, pp. 61-64, doi: 10.1109/IBDAP55587.2022.9907400.
- [23] J. Cambeiro, J. Deantoni and V. Amaral, "Supporting the Engineering of Multi-Fidelity Simulation Units With Simulation Goals," 2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C), Fukuoka, Japan, 2021, pp. 317-321, doi: 10.1109/MODELS-C53483.2021.00053.
- [24] B.Durnyak, I.Hileta, I.Pikh, A.Kudriashova, Y.Petiak Designing a Fuzzy Controller for Prediction of Tactile Product Quality. 1st International Workshop on Di-gital Content & Smart Multimedia, DCSSmart 2019, Lviv, Ukraine, December 23-25, 2019. Pp. 70-81.
- [25] I. Sitova and J. Pecerska, "Process Mining Techniques in Simulation Model Adequacy Assessment," 2019 60th International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS), Riga, Latvia, 2019, pp. 1-4, doi: 10.1109/ITMS47855.2019.8940672.