Gazi Üniversitesi
Fen Bilimleri Dergisi
PART C: TASARIM VE TEKNOLOJİ

Gazi University
Journal of Science
PART C: DESIGN AND TECHNOLOGY

GU J Sci, Part C, 13(3): 1268-1285 (2025)

Recent Reviews on Topology Optimization

Fatma Nur ŞEN[1*], Mithat Kutay CAN[1]

[1]*Turkish Aeronautical Association University, Engineering Faculty, Department of Mechanical Engineering, Ankara, Türkiye*

Article Info

Review article
Received: 09/08/2025
Revision: 22/09/2025
Accepted: 23/09/2025

Keywords

Topology Optimization
Optimization Algorithms
Meta-heuristic Algorithms
Continium Structural Topology

Makale Bilgisi

Derleme makale
Başvuru: 09/08/2025
Düzeltme: 22/09/2025
Kabul: 23/09/2025

Anahtar Kelimeler

Topoloji Optimizasyonu
Optimizasyon Algoritmaları
Sezgi Üstü Algoritmalar
Süreklilik
Yapısal Topolojisi

Graphical/Tabular Abstract (Grafik Özet)

This study represent the literture reviews and comparison of non-gradient based optimization methodologies in terms of structural optimization./ Bu çalışma, yapısal optimizasyon bakımından eğimsiz optimizasyon yöntemlerinin literatür incelemelerini ve karşılaştırılmasını göstermektedir.

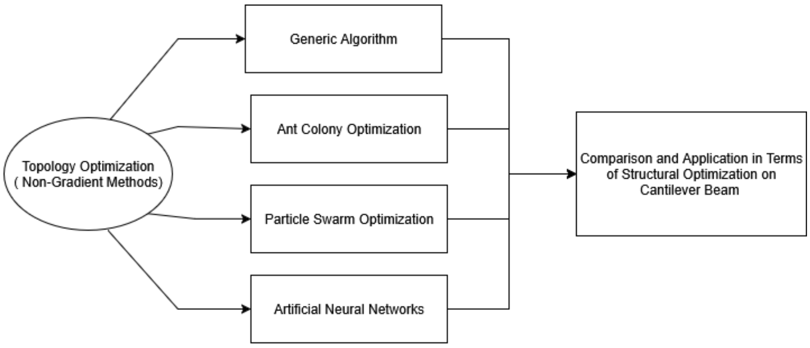


Figure A: *Non-Gradient methods in structural topology optimization: comparative application on cantilever beams /* ***Şekil A:*** *Yapısal topoloji optimizasyonunda gradyansız yöntemler: konsol kirişlerde karşılaştırmalı uygulama*

Highlights (Önemli noktalar)

- *İlk olarak, belirtilen optimizasyon yöntemlerin literatür taraması yapılmıştır. / Initially, the literatur reviews of the specifsied optimization methods have been done.*
- *Daha sonra, bu metotlar yapısal optimizasyon (kiriş) bakımından kıyaslanmıştır. / Then, these methods have been compared in terms of structural optimization (cantilever beam).*
- *Sonuçlara göre, metotlar ve kullanım alanları tablolaştırılmıştır ve araştırmacılara yol gösterici öneri sunmuştur. / Regarding results, methods and their applications were tabulated and they have been presented to resarchers guiding conclusions.*

Aim (Amaç): *This study aims to compare the non- gradient based optimization methods to define the possible best methods for structural optimization of cantilevear beams. / Bu çalışma, kirişin yapısal optimizasyou için mümkün olan en iyi yöntemleri tanımlamak amacıyla eğimsiz optimization yöntemleri karşılaştırmayı amaçlamaktadır.*

Originality (Özgünlük): *This study compounds the informative tabulated contents for the optimization review studies. / Bu çalışma optimizasyon derleme makaleleri için bilgilendirici tablolaştırılmış içerikleri bir araya getirmiştir.*

Results (Bulgular): *The possible best non-gradient optimization methods can be determined for the cantilevear beam. / Kiriş için mümkün olan en iyi eğimsiz optimizsayon yöntemleri belirlenebilir.*

Conclusion (Sonuç): *The important points to be taken into consideration by researchers have beenindicated for future works. / Gelecekteki çalışmalar için araştırmacıların dikkat etmesi gerekenönemli noktalar belirtilmiştir.*

*Corresponding author, e-mail: fnsen@thk.edu.tr
DOI: 10.29109/gujsc.1761253

GU J Sci, Part C, 13(3): 1268-1285 (2025)

Gazi Üniversitesi
Fen Bilimleri Dergisi
PART C: TASARIM VE TEKNOLOJİ

Gazi University
Journal of Science
PART C: DESIGN AND TECHNOLOGY

http://dergipark.gov.tr/gujsc

Recent Reviews on Topology Optimization

Fatma Nur ŞEN[1*], Mithat Kutay CAN[1]

[1]Turkish Aeronautical Association University, Engineering Faculty, Department of Mechanical Engineering, Ankara, Türkiye

Article Info

Review article
Received: 09/08/2025
Revision: 22/09/2025
Accepted: 23/09/2025

Keywords

Topology Optimization
Optimization Algorithms
Meta-heuristic Algorithms
Continium structural topology

Abstract

Gradient-based methods are utilized in traditional topology optimization studies. This approach is based on a point-by-point technique, which depends on the gradient information of the objective functions regarded as independent variables. Despite the fast response, this approach can lead them to find local optimal solutions, but it faces problems in defining the global optimal solutions in large-scale problems or high-degree functions. For this reason, non-gradient methods that give better solutions have been developed to solve these struggles. This study examines the nature-inspired methods developed over the last 25 years, which are called Genetic Algorithms, Ant Colony Optimization, Particle Swarm Optimization, and Artificial Neural Networks, and presents flowcharts to illustrate their principles. These methods are clarified via a cantilever beam. According to the results, methods are compared, and Particle Swarm Optimization can provide a reasonable solution to determine the optimal solution. These comparisons are tabulated and may be a guide for researchers.

Topoloji Optimizasyonunda Güncel Yaklaşımlar

Makale Bilgisi

Derleme makale
Başvuru: 09/08/2025
Düzeltme: 22/09/2025
Kabul: 23/09/2025

Anahtar Kelimeler

Topoloji Optimizasyonu
Optimizasyon Algoritmaları
Sezgi üstü Algoritmalar
Süreklilik yapısal topolojisi

Öz

Geleneksel topoloji optimizasyon çalışmalarında eğimli yöntemler olarak belirtilen metotlar kullanılmaktadır. Bu yaklaşım, bir dizi bağımsız değişkene bağlı olarak amaç fonksiyonun türevine dayanan hesaplamalı nokta tekniğine bağlıdır. Bu metotlar, hızlı çözümler vermesine rağmen, kapsamlı işlemler, yüksek mertebeli fonksiyon gibi durumlarda yerel optimum çözümleri bulmakta; küresel sonuçlarda problemlerle karşılaşılmaktadır. Karşılaşılan sorunları çözmek amaçlı daha iyi sonuçlar elde edilen eğimsiz yöntemler geliştirilmiştir. Bu çalışmada, son 25 yılda yapılan eğimsiz yöntemler olarak belirtilen ve doğadan ilham alan Genetik Algoritmalar, Karınca Kolonisi Optimizasyonu, Parçacık Sürüsü Optimizasyonu ve Yapay Sinir Ağları adlı yöntemler hakkında araştırmalar yapılmış ve çalışma prensipleri akış şeması ile belirtilmiştir. Bu yöntemler, özellikle kiriş örneği ile detaylandırılmıştır. Sonuçlara bağlı olarak, yöntemler karşılaştırılmakta ve Parçacık Sürü Optimizasyonu optimum çözümü belirlemek için uygun bir sonuç sunulabilmektedir. Bu karşılaştırmalara dayanarak, araştırmacılara rehber olabilecek kıyaslama tablosu oluşturulmuştur.

1. INTRODUCTION (GİRİŞ)

Topology optimization addresses a key engineering challenge: determining the optimal distribution of material within a given space to maximize structural performance. Initially developed for mechanical design, this technique has found applications in diverse fields such as fluid dynamics, acoustics, electromagnetism, optics, and multi-physics problems. The process typically involves iterative analysis and design modifications.

Different categories of optimization techniques are sorted as gradient-based methods and non-gradient-based methods. The gradient-based method depends on the direction of the objective function's gradient. This method generally converges quickly and requires fewer function evaluations, particularly when considering the number of independent variables. Despite the advantages, the main drawback of this method is that the gradient-based methods converge to a local optimum and a global

*Corresponding author, e-mail: fnsen@thk.edu.tr
DOI: 10.29109/gujsc.1761253

optimum depending on the initial conditions of the system.

In contrast to the gradient-based methods, the non-gradient-based methods do not depend on the gradient of the function and work based on the function evaluation. This method tries to mimic nature algorithms, and one of the main differences in this method is that the non-gradient-based methods search multiple points in space rather than a single point [1].

Naturally, topology optimization techniques have advanced and evolved due to complex problems and limitations of gradient-based methods. Although non-gradient, nature-inspired methods have struggles and limitations, these methods can be suitable to handle the non-differentiable, discrete or multi-objective problems. Indeed, these methods have strengths in terms of flexibility, global capability, and effectiveness in complex model space.

Topology optimization is a primary approach in structural optimization, aiming to find the optimal structure that meets specific requirements, such as functionality. Furthermore, structural topology can be determined through optimal modifications of connections and holes in the design, as well as the size and shape of the structures, achieved by redistributing material in an iterative method. Therefore, structural topology optimization can be a compelling approach to select the applicable initial structural topologies [2-3].

Topology optimization for structures, as defined as a shape optimization problem, has gained attention in the last decades. Hence, several studies have built upon and improved earlier advancements in homogenization theory and numerical optimization, laying the groundwork for Bendsøe and Kikuchi's influential 1988 work on numerical topology optimization. Since this initial "homogenization approach," the field has evolved in various directions, including: (a) density-based methods; (b) level set methods; (c) methods using topological derivatives; (d) phase-field methods; and (e) evolutionary methods, among others. Density and evolutionary approaches in topology optimization use straightforward design variables linked to elements or nodes. In comparison, the level set method, often paired with topological derivatives, utilizes shape derivatives to determine the optimal topology. Recently, hybrid approaches have emerged, where filtered density fields in advanced projection techniques increasingly mirror the level set function used in level set methods [4-10].

In light of this information, Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Neural Network (NN) methods can be addressed in terms of structural engineering and explained in Section 2. The comparison of these methods is clarified and concluded briefly in Section 3.

2. LITERATURE REVIEW (LİTERATÜR İNCELEMESİ)

2.1. Genetic Algorithms on Topology Optimization (Topoloji Optimizasyonunda Genetik Algoritmalar)

Inspired by natural selection, the Genetic Algorithm (GA) is a search algorithm based on a population that employs the "survival of the fittest" principle. Each new generation is created by using the operators created for this algorithm and iteratively applying these operators to individuals within the population. The core components of a Genetic Algorithm (GA) are chromosome representation, selection, crossover, mutation, and the computation of the fitness function. The GA process follows these steps: Firstly, the Genetic Algorithm (GA) process begins by randomly generating an initial set of chromosomes. Then, the fitness of each chromosome is assessed. Two chromosomes are selected for reproduction based on their fitness scores in the third step. Indeed, these selected chromosomes undergo a single-point crossover process with a certain probability to create an offspring. Finally, one operator of the genetic algorithm, named mutation, is applied evenly to this offspring with a specific mutation chance, resulting in a mutated offspring. This mutated offspring is added to the new population. These steps of selection, crossover, and mutation are repeated until the new population is fully formed. Given that GA can provide a diverse set of solutions, it can be used in many engineering areas such as structural operations, control engineering, computer games, artificial intelligence, deep learning, and image processing [11-14]. The flow chart for the Genetic Algorithm can be seen in Figure 1.

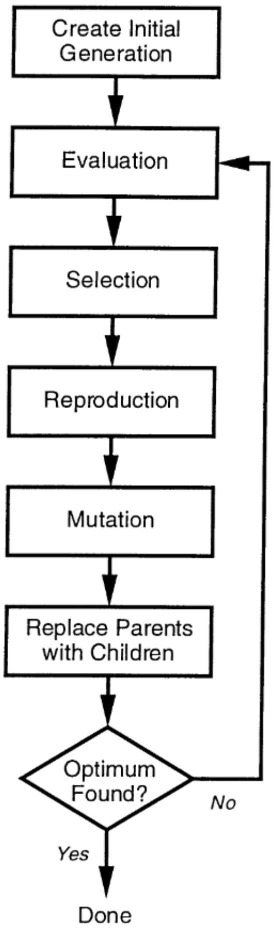


Figure 1. Flow chart of Genetic Algorithm [17] (Genetik Algoritmanın Akış Şeması [17])

In terms of structural optimization, several studies have been conducted over the last decades [2, 15-21]. Based on these studies, structural optimization started with the truss optimization. Ohsaki's study [15] indicates the optimal topologies of truss systems under static loading conditions. Indeed, singularity problems due to the zero cross-sectional area are overcome with GA methods. The cross-section of every member was designed as a string of binary data. Firstly, initial values were determined randomly, and the fitness value was defined via a function. Then, the other processes, mutation and crossover, are applied to find an optimal solution. Finally, parents were selected, and children were produced until the optimal solution was achieved. However, this algorithm can be affordable for large truss systems. Another study [17] addressed that the traditional ground structure method was compensated for by GA methods optimization. In this study, the selection of penalty functions was essential to avoid different local optimums in each new population. Numerical examples were simulated with varying numbers of nodes. As an improvement, this method can be integrated into rigid jointed structures. According to [2], topology optimization for beam and plate examples involves a combination of graph representation of geometrical structures and ranking the results using fitness assignment and stochastic sampling. The procedure begins by creating a graph representation for the genetic algorithm to work on. For their first example in [2], a cantilever plate with the left side of the plate fixed on the wall and a force applied vertically at the right side of the plate at the midpoint is used. Figure 2 shows the mesh and graphical representations of the cantilever plate as shown in Figure 3.

Using these graph representations and genetic algorithms, the constrained optimization for an artificial unconstrained objective function will be calculated, and at the end, the optimized topology result will be found. To achieve this result, a penalty function approach is necessary to generate feasible individuals within the population. The optimized topology for the cantilever plate, as shown in Figures 4 and 5, corresponds to the two different graph representations in Figure 3.

As a conclusion, in all studies [15-21], the Genetic Algorithm method was applied as a novel method, especially in contrast to the ground structure topology method, and they reached advantages in the GA method. Furthermore, the selection of the penalty and fitness function becomes crucial to take these advantages.

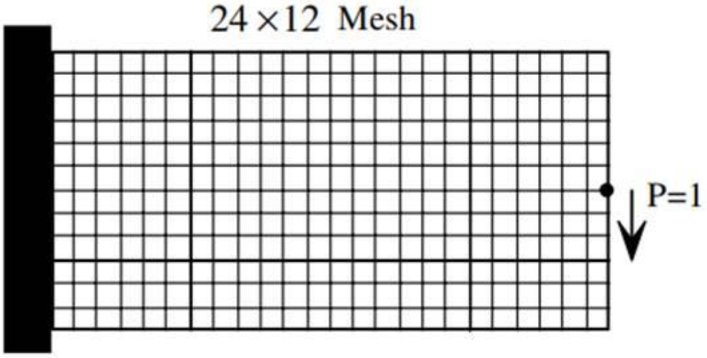


Figure 2. Cantilever plate mesh used for the optimization [2] (Optimizasyon için kullanılan konsol plaka ağı [2])

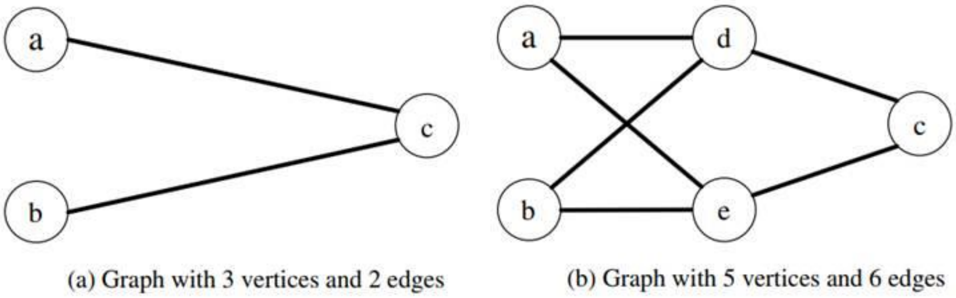


Figure 3. Examples of two graph representations used for minimum compliance design [2] (Minimum uyumluluk tasarımı için kullanılan iki grafik gösteriminin örnekleri [2])

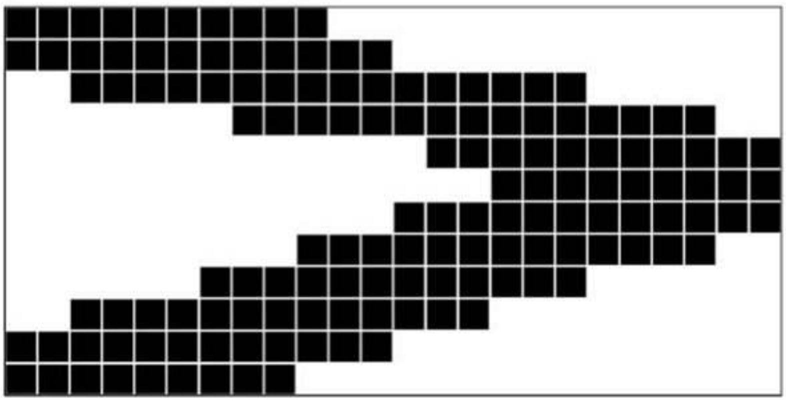


Figure 4. Three-vertex two-edge graphical representation based optimization solution [2] (Üç köşeli iki kenarlı grafiksel gösterime dayalı optimizasyon çözümü [2])

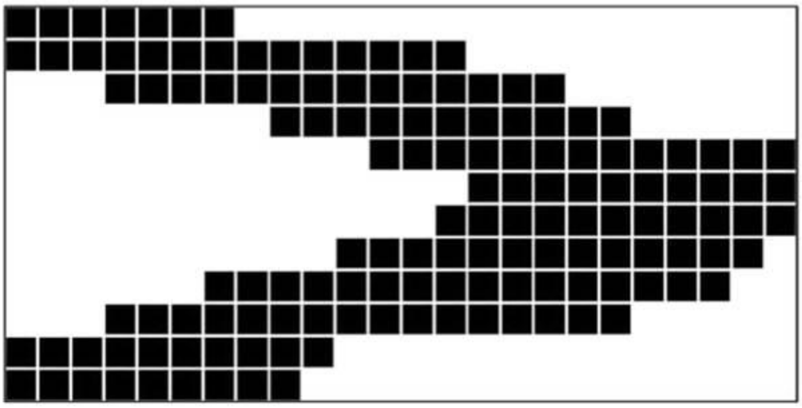


Figure 5. Five-vertex and six-edge graphical representation based optimization solution [2] (Beş köşeli ve altı kenarlı grafiksel gösterime dayalı optimizasyon çözümü [2])

2.2. Ant Colony Optimization on Topology Optimization (Topoloji Optimizasyonunda Karınca Kolonisi Optimizasyonu)

Ant Colony Optimization (ACO) uses a group of computer-created ants to find the optimum point for a given optimization problem. These virtual ants then share information about the quality of their solutions through a communication mechanism analogous to how real ants communicate [22]. The proposed solution follows a structured sequence comprising Initialization, Initial Design Generation, Solution Component Selection, Analysis and Evaluation, Cycle Completion Verification, and Pheromone Update. Initializations start with a certain number of sections, and a certain number of paths are selected. A pheromone level characterizes each path. This level indicates the suitability of the path. These level of phenomena depends on the number of design variables. The next step is constructing the colony of ants. Each of these ants create a solution of its own. Each time an ant follows a path, the pheromone level is decreased. When all the ants select a path and follow, the probabilities of the paths are recalculated. Indeed, using an objective function, the results of all ants are calculated and compared. An optimal local search can also be applied to the solutions. After these pheromone levels around the path s will be updated. These updates can be evaporation or reinforcement. These can be selected if we have any specific rules; more high-quality solutions depend on the objective function result, and to reduce uniformly to avoid unlimited accumulation and to encourage exploration [23-24]. The flow chart for the ant colony optimization is shown in Figure 6. These processes are repeated until a stagnation point is reached or predefined criteria are met. This optimization can be used in civil engineering, timetabling, data mining, cell placement in circuit designs, the vehicle routing problem, and even in music [25-30]. Regarding the scope of this study, there are several studies in structural topology [31-34].

Hassani et. al [31] used an ant colony optimization algorithm for topology optimization. Real ants can locate the shortest path between the nutrient source and their nest by relying on a chemical substance called pheromones, rather than visual cues. This behavior has inspired the creation of Ant Colony Optimization (ACO), a method in which a group of computers creates ants that work together to address a combinatorial problem by transmitting indirect information through an artificial pheromone trail. This trail builds up over time through a learning process that rewards effective solutions to the problem. For example, one of the cases uses a cantilever beam with boundary conditions shown in Figure 7. Lastly, the findings of optimal topology are shown in Figure 8.

Luh et. al [32] explored the classic cantilever beam problem using a modified Ant Colony Optimization (ACO) algorithm. The goal was to find the most efficient structures by minimizing weight while maintaining sufficient stiffness. The design space was broken down into finite elements, and the algorithm, much like a colony of ants, built up various designs, element by element. When subjected to four distinct loading scenarios, the method did not yield a single perfect answer. Instead, it produced a wide array of functional topologies, each representing a unique balance of stiffness and weight. The resulting designs were not just different in shape but also in their internal makeup, showing both truss-like and more composite-like structures. This demonstrates the algorithm's power in generating diverse solutions, providing designers with a valuable set of lightweight alternatives tailored to specific performance or manufacturing needs.

In their respective studies on using Ant Colony Optimization (ACO) for structural design, Kaveh et al. [31] and Luh and Lin [32] approached the problem with different goals. Kaveh focused on a practical, clear-cut method to achieve the stiffest possible structure for a given amount of material. Their approach treated the design problem as a straightforward on-or-off decision for each element, which avoids the ambiguous "grey areas" seen in other methods. They even added a technique to remove non-functional parts, resulting in designs that are both realistic and easy to build. In contrast, Luh and Lin took a more conceptual and exploratory path. Instead of seeking a single optimal design, they aimed to discover a whole variety of high-quality solutions. They improved the basic ACO algorithm with features like "elitist ants" and "multiple-colony memories" to encourage the creation of diverse and innovative designs. Their work highlights that many reasonable solutions can exist for a single design problem, offering engineers a range of viable and creative options. Essentially, while Kaveh et al. delivered a direct and efficient way to create a manufacturable design, Luh and Lin demonstrated how to evolve the ACO method itself to foster diversity and robustness in generating structural topologies.

2.3. Particle Swarm Optimization on Topology Optimization (Topoloji Optimizasyonunda Parçacık Sürüsü Optimizasyonu)

Particle Swarm Optimization is a stochastic optimization method that utilizes the collective behavior of animal groups, such as insects, herds, birds, and fish. In other words, PSO simulates how these swarms cooperate to find resources, with each member adjusting its search pattern based on its own experiences and those of its peers. The main design principles of this method draw from two main areas: evolutionary algorithms, which share the use of a swarm to explore a wide area of the objective function's solution space concurrently, and artificial life, focusing on the study of artificial systems exhibiting life-like characteristics [35,36]. The algorithm structure is sorted into eight parts. These parts are mainly related to adopting populations, developing selection strategies, modifying position or velocity strategies, and maintaining population diversity. These steps are shown in Figure 9 as a flow chart. Furthermore, parameters such as inertia weight, learning factors, position and speed limits, and initial and size of swarm are crucial in this algorithm. Although the PSO is a recent method, it garnered significant attention in many areas. Particle swarm optimization can be used in engineering, healthcare, environmental, industrial, and commercial applications [37-44].

In terms of structural optimization [45,46], based on the study [45], particles refer to a set of arbitrary vectors generated as a solution to the problem. The swarm flies or goes over the domain of the problem, and particles detect a new location or paths in each step. In this way, the quality of locations can be related to the defining fitness function properly. However, in traditional methods, the swarm can be located for the new location, close to the best particle location, and kept in memory. This situation can affect the essential points in the tracking path in the current particle, and local minimums can be trapped. Therefore, the weighted particle can be determined for the new particles and adapt the velocity formulation. Regarding numerical examples, results indicated that the integrated PSO eliminates the drawbacks and can be a sufficient method for discrete and continuous systems.

Luh et al.'s study [46] introduces a modified binary particle swarm optimization (BPSO) algorithm specifically for continuous structural topology optimization. The algorithm translates its binary decisions into a physical design using a special mapping scheme. It begins with a randomized set of designs. It evaluates them based on a multi-objective fitness function that considers stiffness, weight, and stress, while also penalizing designs that punish constraints. The method uses adaptive rules to balance exploring the design space with refining current solutions, and a multi-swarm memory to save diverse good designs, preventing it from converging on just one answer too early. When tested on standard cantilever plate problems, the proposed BPSO proved effective, generating a wide variety of structurally sound and efficient designs, including truss-like layouts. Its improved diversity and solution quality, when compared to other optimization methods, confirm its effectiveness as a robust tool for this type of structural design.

The particle's position and velocity correspond to the genotype parameters, while the binary positions correspond to the phenotype parameter. These binary positions are utilized to map the structure of the topology optimization problem. Figure 10 shows how the binary particle to topology mapping is done.

Both studies apply swarm intelligence to structural optimization, but they target different types of problems and use different enhancements. Luh [46] adapted Particle Swarm Optimization (PSO) to a binary framework to handle continuous topology optimization. Their main contribution is a robust way to translate the algorithm's binary decisions into a physical design, allowing it to efficiently explore different material layouts and produce a wide range of feasible solutions. In contrast, Mortazavi and Toğan [45] extend PSO to a more complex problem involving truss systems. Their approach simultaneously optimizes the size, shape, and topology of the truss. They introduce specialized techniques, such as weighted particles and rules, which help them deal with constraints, thereby improving the algorithm's ability to find feasible solutions and converge effectively. In short, [46] work addresses the challenge of making PSO work for binary design spaces in continuous structures, while [45] work tackles the mixed continuous and discrete variables found in truss optimization. Together, they demonstrate the versatility of PSO.

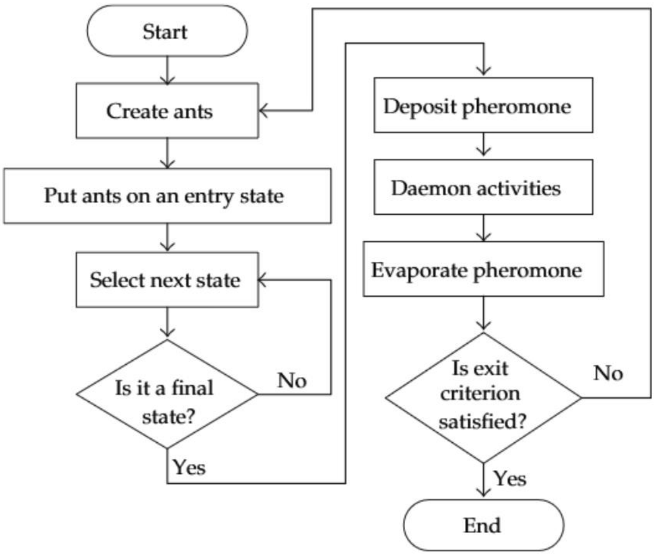


Figure 6. Ant colony optimization flowchart [31] (Karınca kolonisi optimizasyon akış şeması [31])

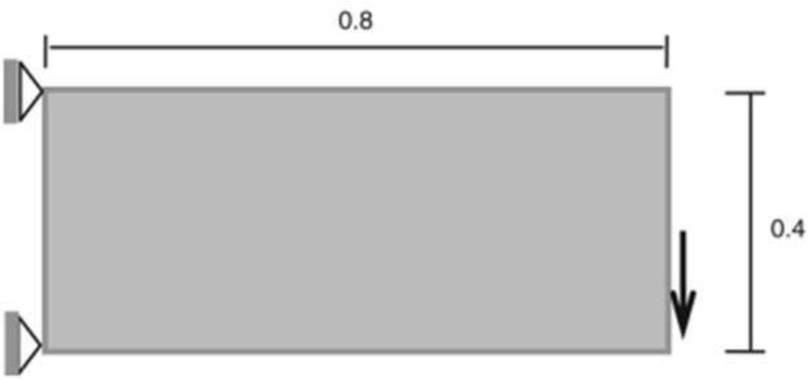


Figure 7. Cantilever beam problem defined in [31] (Tanımlanan konsol kiriş problemi [31])

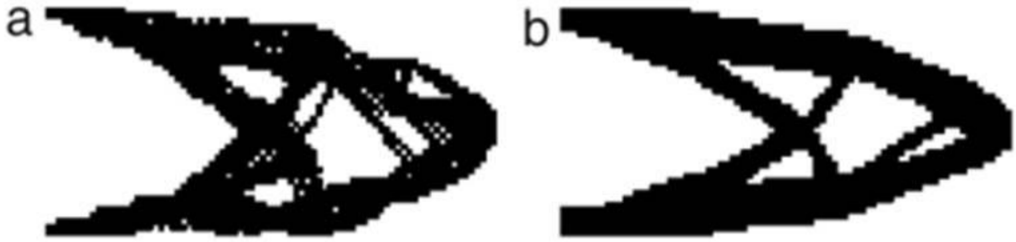


Figure 8. Final results of the optimization including the nose and not respectively [31] (Burun ve burun olmayan kısımları içeren optimizasyonun nihai sonuçları [31])

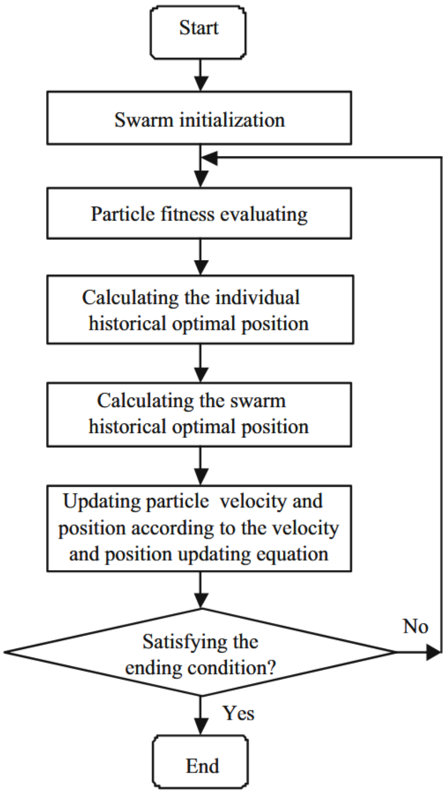


Figure 9. Particle swarm optimization flowchart [35] (Parçacık sürüsü optimizasyon akış şeması [35])

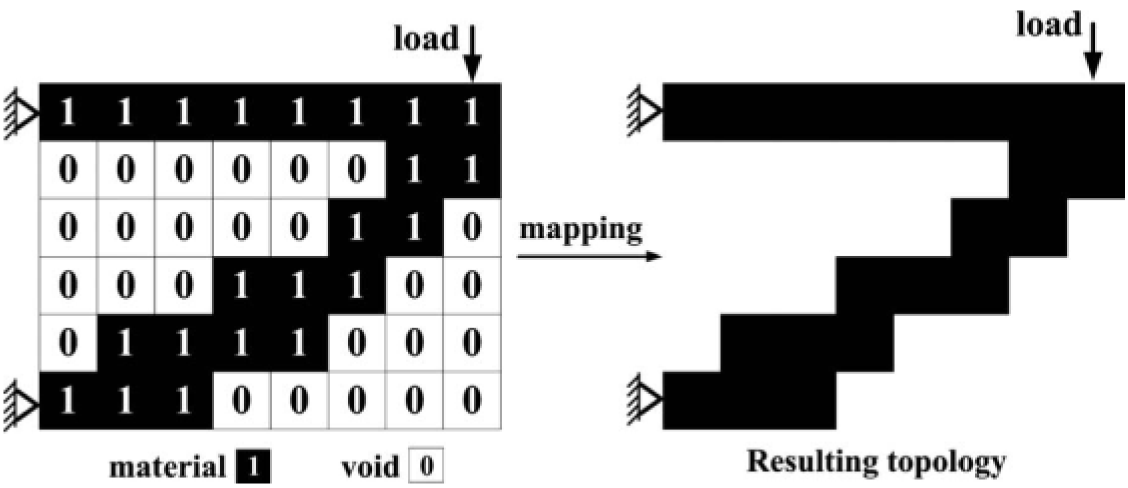


Figure 10. Binary particle domain method-based mapping for topology (Topoloji için ikili parçacık alanı yöntemi tabanlı eşleme)

2.4. Neural Network on Topology Optimization
(Topoloji Optimizasyonu Sinir Ağı)

The human nervous system contains a vast network of interconnected neurons, which are called nerve cells. The term "neural" is an adjective derived from "neuron," while "network" refers to a graph-like structure. Artificial Neural Networks (ANNs) are computing systems that draw inspiration from the biological neural networks found in humans. These systems are also known as "Neural Nets," "parallel distributed processing systems," or "connectionist systems [47,48].

To be classified as an ANN, a computing system must have a directed graph structure. In this structure, the nodes (or vertices) perform simple computations, and the connections (or edges/links/arcs) link pairs of these nodes. This structure is a fundamental concept from elementary graph theory, which defines a directed graph as a set of nodes and the connections between them [47]. Neural networks can be used for speech recognition, several engineering areas, computer vision, and pattern recognition [47-53].

Banga et al. [54] work on using a deep learning method based on a 3D encoder-decoder Convolutional Neural Network to improve the solutions of the Solid Isotropic Material with Penalization process. The proposed method first uses SIMP to perform iterations to obtain an intermediate solution. Then, using the neural network (NN) based on density distribution, the NN creates the results. From the results, the NN calculated the same result with % 40% less computation. In Figure 11, the results are shown. The left side shows the structure and the forces acting on it. The right side shows the CNN prediction and the results of the optimization program.

Meng et al. [55] use a neural network based on finite element analysis, which focuses on solving reliability-based topology optimization. Again, this article focuses on decreasing computational time. The method uses numerical basic functions to calculate the elements on the mesh. The neural network-based deep learning algorithm does each numerical calculation. Then the results were analyzed with displacement solutions and sensitivity analyses. With these results and numerical solutions, the optimal density distribution is created. Since this is a reliability-based topology optimization, the displacement function contains a reliability index parameter; by changing this parameter, different results can be achieved, and the results are shown in Figure 12.

Chandrasekhar and Suresh [56] work on a method for topology optimization (TO) via neural networks (NN). The key idea is to utilize the NN's activation functions to represent the material density field, which is typically managed by a method called Solid Isotropic Material with Penalization (SIMP). Regarding this approach, the density function is defined via the weights and biases of the NN, making it independent of the finite element mesh. The optimization process relies on the NN's backpropagation algorithm, along with a standard finite element solver, to adjust the density field and find the optimal design. Their results, compared with the regular optimization programs' results, are shown in Figure 13.

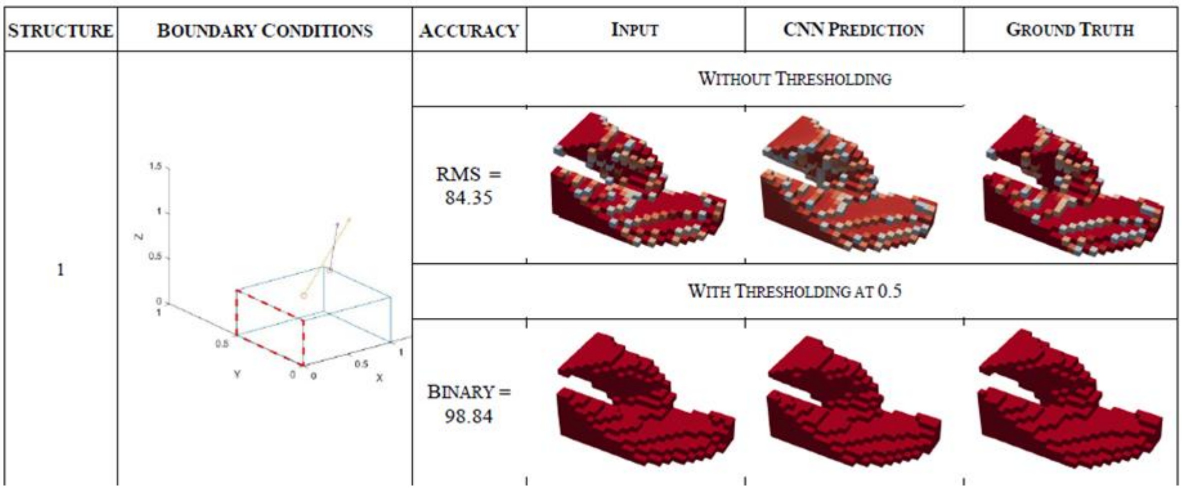


Figure 11. Convolutional Neural Network vs TopOpt Results Comparison with Boundary conditions [54]
(Sınır koşullarıyla Evrişimli Sinir Ağı ve TopOpt Sonuçlarının Karşılaştırılması [54])

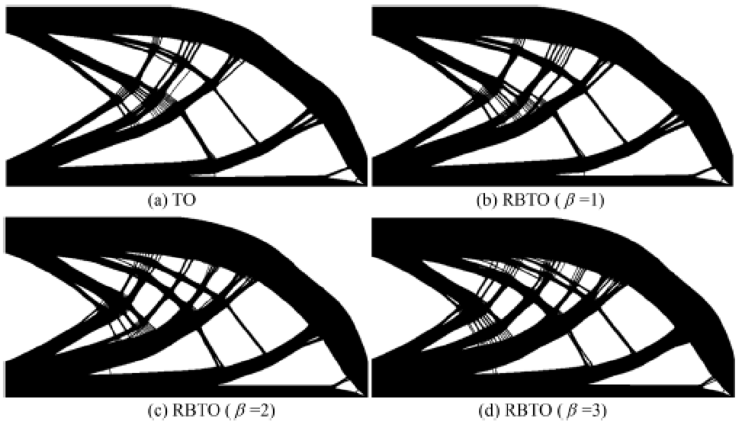


Figure 12. RBTO results versus normal Topology Optimization results [55] (RBTO sonuçları ve normal Topoloji Optimizasyonu sonuçları [55])

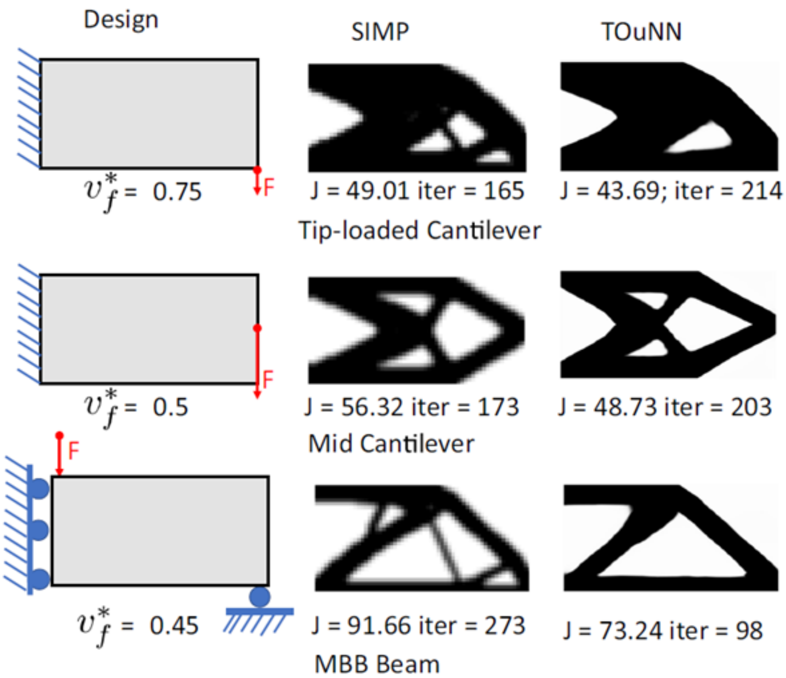


Figure 13. Different boundary conditions optimized comparison between the proposed method and the general optimization program results (Önerilen yöntem ile genel optimizasyon programı sonuçları arasında farklı sınır koşulları optimize edilerek karşılaştırma yapılmıştır)

Xing's [57] article introduces a new method called stochastic gradient online learning and prediction (SGoLap) to speed up structural topology optimization. Moreover, this method uses a one-hidden-layer recurrent neural network (RNN) to learn and predict derivative information, including the second-order derivative of the objective function, as optimization progresses. By doing this, it can selectively skip some of the more computationally expensive steps, such as finite element analysis (FEA) and sensitivity analysis, which saves total computational time. Two different prediction methods based on neural networks are used. The results are shown in Figure 14. The figure compares a standard compliance solving algorithm with the prediction methods from the article, presenting final objective values and their relative differences.

Each of these four works uses neural networks for topology optimization, but their approaches and goals are different. [56] proposes a fundamental change by using neural network activation functions to directly define the density of the material, which creates a design that is not dependent on the finite element mesh and is also fully differentiable. Meng [55] applies this to the field of reliability-based optimization, incorporating uncertainty into their method by combining DNNs with a specialized finite element method to ensure the resulting designs are robust even for enormous and complex problems. In a more practical vein, Banga [54] focuses on speeding up the design process. They use convolutional neural networks (CNNs) that have been pre-trained on intermediate design steps from a standard SIMP optimization, allowing them to quickly generate final designs with a good balance of speed and accuracy. Finally, Xing and Tong [57] introduce an online, recurrent neural network (RNN)-based method. This approach learns as it goes, predicting the necessary gradient and Hessian information in real-time. This dramatically reduces the computational cost of the optimization process itself, eliminating the need for extensive pre-training. Taken together, these studies represent a broad range of applications for neural networks in topology optimization: from creating a new direct formulation [56], to handling uncertainty [55], to speeding up the process using pre-trained data [54], and to accelerating the optimization process in real-time [57].

In this article, Genetic algorithms (GA), Neural Networks (NN), Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO) are focused on optimizing the topology of materials used in any design based on the force and support types and placements. A comparison table between the algorithms is given in Table 1. Nature-inspired optimization algorithms provide flexible and powerful approaches for addressing complex challenges across critical domains such as healthcare, engineering, and information technology. Techniques like Artificial Neural Networks (ANNs) demonstrate strong capabilities in pattern recognition, enabling applications in areas such as medical diagnosis, while Ant Colony Optimization (ACO) offers efficient solutions to routing and logistics-related problems. At the same time, methods such as Genetic Algorithms (GAs) and Particle Swarm Optimization (PSO) are extensively utilized for design and scheduling tasks, with particular effectiveness in advanced engineering applications, including structural topology optimization. Indeed, the comparison of applications of algorithms can be shown in Table 2 [58- 60].

3. DISCUSSION & CONCLUSIONS (TARTIŞMA & SONUÇLAR)

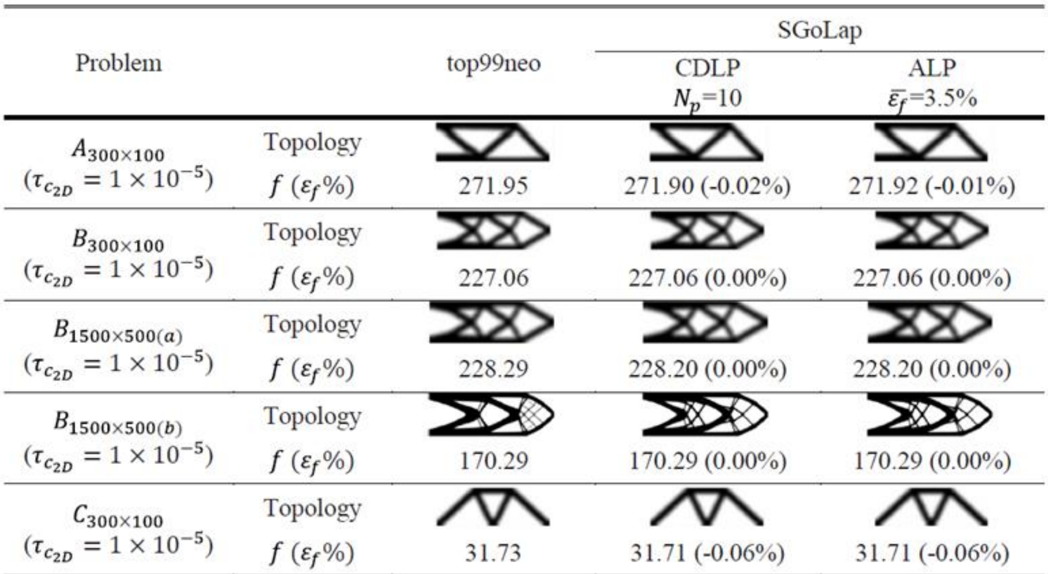
In this This study reviews non-gradient methods inspired by nature, including Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Artificial Neural Network Optimization (ANN). Although these methods were examined only for structural optimization with a cantilever beam, they can be applied in many engineering areas.

Firstly, non-gradient-based methods show a better optimal solution than gradient-based methods because the objective is nondifferentiable, and their objective functions can be multiple, such as fitness and weight.

Among the four methods, GA can handle the discrete or multi-objective problems and is suitable for finding global optima. However, it can converge more slowly than PSO and ACO. PSO methods can be ideal for structural problems involving continuous structures. Nevertheless, the traditional PSO may be trapped in a local minima problem, and position and velocity constants may be modified. For the ACO, this method can be effective for discrete problems or truss systems and is better for determining short load paths. The drawback of the method is that its performance heavily depends on the update rules of the pheromone, and ACO is slower than PSO. In a Neural Network, this method can be used to guide the optimization and help run many optimization cases. Indeed, training is one of the critical issues. After training, an ANN can predict quickly, but a wide data repository is a significant concern for training, which may lead to cost concerns. Another drawback of ANN is that it can be less interpretable due to the complex internal structure and distributed representations. Finally, Neural Networks can be used in cases where real-time design assistance is necessary. In light of this comparison, Particle Swarm Optimization can be selected as the most effective non-gradient method in terms of performance and computation in the beam example.

Although gradient-based methods are typically favored for their efficiency in problems where gradients are easily calculated, non-gradient algorithms offer valuable flexibility and robustness when gradients are complicated or unreliable to compute, such as in overly complex or discrete design spaces. Gradient and non-gradient-based methods can be used simultaneously to overcome difficulties. This hybrid approach could improve the robustness of the solution process and yield more detailed, accurate, and versatile results in topology optimization, especially in applications with complex design constraints or uncertain material properties.

As future work, these methods will be integrated into each other for design and optimization purposes. Another possible area of work is to improve the efficiency and adaptability of methods in complex engineering and real-world problems, as these methods, especially GA and ACO, can face slow convergence rate and computational cost concerns. Therefore, both methods will be integrated with machine learning techniques as they can provide more balance between global and local exploitation. Moreover, adaptive control mechanisms and self-tuning approaches will be crucial to enhance the flexibility and performance in diverse problems.



Problem		top99neo	SGoLap CDLP N_p=10	SGoLap ALP $\bar{\varepsilon}_f$=3.5%
$A_{300\times100}$ ($\tau_{c_{2D}} = 1\times10^{-5}$)	Topology			
	f (ε_f%)	271.95	271.90 (-0.02%)	271.92 (-0.01%)
$B_{300\times100}$ ($\tau_{c_{2D}} = 1\times10^{-5}$)	Topology			
	f (ε_f%)	227.06	227.06 (0.00%)	227.06 (0.00%)
$B_{1500\times500(a)}$ ($\tau_{c_{2D}} = 1\times10^{-5}$)	Topology			
	f (ε_f%)	228.29	228.20 (0.00%)	228.20 (0.00%)
$B_{1500\times500(b)}$ ($\tau_{c_{2D}} = 1\times10^{-5}$)	Topology			
	f (ε_f%)	170.29	170.29 (0.00%)	170.29 (0.00%)
$C_{300\times100}$ ($\tau_{c_{2D}} = 1\times10^{-5}$)	Topology			
	f (ε_f%)	31.73	31.71 (-0.06%)	31.71 (-0.06%)

Figure 14. Comparison results in different cases between common algorithm the new algorithm [57] (Ortak algoritma ile yeni algoritma arasındaki farklı durumlarda karşılaştırma sonuçları [57])

Table 1. The comparison of non-gradient-based methods (Gradyan tabanlı olmayan yöntemlerin karşılaştırılması)

Method	Genetic Algorithm (GA)	Neural Networks (NN)	Ant Colony Optimization (ACO)	Particle Swarm Optimization (PSO)
Inspiration	Natural evolution	Learning Methods of the Human Brain	Ant search, explore, and exploit methods	Behaviour patterns of bird/fish swarms
Search Strategy	**Exploration** (mutation) and **exploitation** (crossover and selection) of a population of solutions.	Gradient-based or other optimization methods (like GA or PSO) to adjust weights and biases.	**Positive feedback** based on pheromone trails, leading to a focus on promising paths.	Particles move through the search space, influenced by their own best-found position and the best position found by the entire swarm.
Problem Types	Combinatorial, discrete, and continuous optimization; practical for complex, non-linear problems.	Supervised, unsupervised, and reinforcement learning; classification, regression, and pattern recognition.	Combinatorial optimization is well-suited explicitly for shortest path problems on graphs.	Continuous optimization; also used for discrete and combinatorial problems.
Strengths	**Robust** for complex, high-dimensional problems. Avoids getting stuck in local optima. Can handle non-differentiable and non-linear functions. Flexible and easy to hybridize.	Exceptional at **pattern recognition** and function approximation. Can learn and generalize from data. Highly effective for complex, non-linear relationships.	Good at finding near-optimal solutions for **path-finding** and routing problems. Naturally parallelizable. Robust to changes in the problem environment.	**Fast convergence** and computationally efficient. Simple to implement. Requires fewer parameters to tune compared to GA. Can handle high-dimensional spaces.
Weaknesses	Can be computationally expensive and slow to converge. Requires careful tuning of parameters (e.g., population size, mutation rate). No guarantee of finding the global optimum.	Requires large amounts of data for training. Prone to getting stuck in local minima (without additional optimization). **The black box** nature makes it hard to interpret the internal logic.	Slower convergence for some problems compared to PSO. Can suffer from stagnation if pheromone values are not managed correctly.	Prone to **premature convergence**, where the entire swarm moves toward a sub-optimal solution. Less effective for problems with many local optima.

Table 2. The comparison between optimization methods and application areas (Optimizasyon yöntemleri ve uygulama alanları arasındaki karşılaştırma)

Algorithm Type	Healthcare & Medicine	Engineering & Manufacturing	Information Technology	Transportation & Logistics
Artificial Neural Networks (ANN)	Analyzing medical scans, diagnosing diseases, discovering new drugs and creating personalized treatment plans based on genomic data.	Improving product quality, predicting equipment failure (predictive maintenance), controlling industrial processes, and operating robotics.	Recognizing patterns in data, understanding human language (NLP), computer image analysis, and strengthening cybersecurity defenses.	Forecasting traffic conditions, enabling autonomous vehicle navigation, optimizing travel routes, and predicting demand for services.
Particle Swarm Optimization (PSO)	Pinpointing regions in medical images, developing optimal treatment plans, and fine-tuning drug dosages for maximum effectiveness.	Enhancing structural and system designs, tuning control systems for better performance, and optimizing the efficiency of power systems.	Selecting the most relevant features in a dataset, clustering data, training other neural networks, and conducting software testing.	Planning the most efficient routes for vehicles, optimizing traffic light timing to reduce congestion, and managing entire fleets of vehicles.
Genetic Algorithms (GA)	Aiding in the design of new pharmaceuticals, creating optimal schedules for treatments and equipment use, and analyzing genetic sequences.	Designing electronic circuits and mechanical parts, optimizing manufacturing schedules, and improving overall system architecture.	Automating software testing, improving code efficiency, selecting key data features for analysis, and configuring complex systems.	Solving complex vehicle routing challenges, designing efficient logistics networks from the ground up, and scheduling transportation tasks.
Ant Colony Optimization (ACO)	Efficiently allocating hospital resources (like staff and beds), managing medical supply chains, and finding the best treatment pathways for patients.	Scheduling tasks in a manufacturing environment, making assembly lines more efficient, and planning preventative maintenance for machinery.	Optimizing data routing in computer networks, discovering insights from large datasets (data mining), and managing tasks in distributed computing.	Finding the best routes for delivery vehicles, designing transportation and logistics networks, and managing real-time traffic flow.

DECLARATION OF ETHICAL STANDARDS
(ETİK STANDARTLARIN BEYANI)

The author of this article declares that the materials and methods they use in their work do not require ethical committee approval and/or legal-specific permission.

Bu makalenin yazarı çalışmalarında kullandıkları materyal ve yöntemlerin etik kurul izni ve/veya yasal-özel bir izin gerektirmediğini beyan ederler.

AUTHORS' CONTRIBUTIONS (YAZARLARIN KATKILARI)

***Fatma Nur ŞEN*:** She conducted the literature review, research, editing and writing process.

Literatür taraması, araştırma, düzenleme ve yazma sürecini yürütmüştür.

***Mithat Kutay CAN* :** He conducted the literature review, research, editing and writing process.

Literatür taraması, araştırma, düzenleme ve yazma sürecini yürütmüştür.

CONFLICT OF INTEREST (ÇIKAR ÇATIŞMASI)

There is no conflict of interest in this study.

Bu çalışmada herhangi bir çıkar çatışması yoktur.

REFERENCES (KAYNAKLAR)

[1] Mooneghi, M. A., Kargarmoakhar, R. Aerodynamic mitigation and shape optimization of buildings: review. *Journal of Building Engineering*, 6, (2016), 225-235. doi: 10.1016/j.jobe.2016.01.009

[2] Wang, S. Y., & Tai, K. Graph representation for structural topology optimization using genetic algorithms. *Computers & Structures*, *82*(20–21), (2004), 1609–1622. doi: 10.1016/j.compstruc.2004.05.005

[3] Ahmed, F., Deb, K., Bhattacharya, B., Structural topology optimization using multi-objective genetic algorithm with constructive solid geometry representation. *Applied Soft Computing*, 39, (2016), 240-250. doi: 10.1016/j.asoc.2015.10.063

[4] Bendsøe, M.P. Optimal shape design as a material distribution problem. *Structural Optimization* 1, 193–202 (1989). doi: 10.1007/BF01650949

[5] Zhou, M., & Rozvany, G. I. N. The COC algorithm, part II: Topological, geometrical and generalized shape optimization. *Computer Methods in Applied Mechanics and Engineering*, *89*(1–3),(1991), 309–336. doi: 10.1016/0045-7825(91)90046-9

[6] Allaire, G., Jouve, F., & Toader, A.-M. A level-set method for shape optimization. *Comptes Rendus. Mathématique*, *334*(12), (2002),1125–1130. doi: 10.1016/s1631-073x(02)02412-3

[7] Wang, M. Y., Wang, X., & Guo, D.. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, *192*(1–2), (2003), 227–246. doi: 10.1016/s0045-7825(02)00559-5

[8] Jan Sokolowski, Antoni Zochowski. On Topological Derivative in Shape Optimization. [Research Report] RR-3170, INRIA. (1997), pp.31.

[9] Bourdin, B., & Chambolle, A. Design-dependent loads in topology optimization. *ESAIM: Control, Optimisation and Calculus of Variations*, *9*, (2003) 19–48. doi: 10.1051/cocv:2002070

[10] Xie, Y. M., & Steven, G. P. A simple evolutionary procedure for structural optimization. *Computers & Structures*, *49*(5), (1993), 885–896. doi: 10.1016/0045-7949(93)90035-c

[11] Katoch, S., Chauhan, S.S. & Kumar, V. A review on genetic algorithm: past, present, and future. *Multimed Tools Appl* 80, (2021), 8091–8126. doi: 10.1007/s11042-020-10139-6

[12] A. Lambora, K. Gupta and K. Chopra, "Genetic Algorithm- A Literature Review," *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, Faridabad, India, 2019, pp. 380-384, doi: 10.1109/COMITCon.2019.8862255.

[13] Lingaraj, Haldurai. A Study on Genetic Algorithm and its Applications. In*ternational Journal of Computer Sciences and Engineering*, 4, (2016), 139-143.

[14] De Jong, K. Learning with genetic algorithms: An overview. *Mach Learn* 3, (1988), 121–138. doi: 10.1007/BF00113894

[15] Ohsaki, M., Genetic algorithm for topology optimization of trusses, *Computers&Structures*, 57(2), (1995), 219-225.

[16] Wang, S.Y., Tai, K., Wang, M, Y., An enhanced genetic algorithm for structural topology optimization. I*nt. J. Numer. Meth. Engng*, 65, (2006), 18-44. doi : 10.1002/nme.1435

[17] Chapman, C.D., Structural topology optimization via the genetic algorithm, Master Thesis, Massachusetts Instıtute of Technology, Massachusetts, (1994).

[18] Kaveh, A., Kalatjari, V., Topology optimization of trusses using genetic algorithm,

force method and graph theory. I*nt. J. Numer. Meth. Engng*, 58, (2003), 771-791. doi: 10.1002/nme.800

[19] Kawamura, H., Ohmori, H. & Kito, N. Truss topology optimization by a modified genetic algorithm. *Struct Multidisc Optim* 23, (2002), 467–473. doi: 10.1007/s00158-002-0208-0

[20] Aguilar Maderia, J.F., Rodrigues, H. Pina, H., Multi-objective optimization of structures topology by genetic algorithms. *Advances in Engineering Software*, 36, (2005), 21-28. doi: 10.1016/j.advengsoft.2003.07.001

[21] Wang, S.Y., Tai, K., Structural topology design optimization using Genetic Algorithms with a bit-array representation, Comput. Methods Appl.Mech. Engrg, 194, (2005), 3749-3770. doi:10.1016/j.cma.2004.09.003

[22] Dorigo, M., Birattari., M., Stutzle, T., Ant colony optimization. *IEEE Computational Intelligence Magazine*, 1(4), 2006, 28-39. doi: 10.1109/MCI.2006.329691

[23] Dorigo, M.,Blum, C., Ant colony optimization theory: A survey, *Theoretical Computer Science*,344(2),(2005), 243-278. doi: 10.1016/j.tcs.2005.05.020.

[24] Hasançebi, O. & Çarbas, S., Ant colony search method in practical structural optimization. *Int. J. Optim. Civil Eng*, 1, (2011), 91-105.

[25] Socha, K., Sampels, M., Manfrin, M. Ant Algorithms for the University Course Timetabling Problem with Regard to the State-of-the-Art. In: Cagnoni, S., *et al.* Applications of Evolutionary Computing. EvoWorkshops 2003. Lecture Notes in Computer Science, 2611. Springer, (2003), Berlin, Heidelberg. doi: 10.1007/3-540-36605-9_31

[26] Parpinelli, R. S., Lopes, H. S., & Freitas, A. A., Data mining with an ant colony optimization algorithm. *IEEE Transactions on Evolutionary Computation*, *6*(4), (2002), 321–332. doi: 10.1109/tevc.2002.802452

[27] Alupoaei, S., & Katkoori, S. Ant Colony System Application to macrocell overlap removal. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *12*(10), (2004), 1118–1123. doi: 10.1109/tvlsi.2004.832926

[28] L. M. Gambardella, E. Taillard, and G. Agazzi. 1999. MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. Technical Report. Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale.

[29] Guéret, C., Monmarché, N., Slimane, M. Ants Can Play Music. In: Dorigo, M., Birattari, M., Blum, C., Gambardella, L.M., Mondada, F., Stützle, T. (eds) Ant Colony Optimization and Swarm Intelligence. ANTS 2004. Lecture Notes in Computer Science, vol 3172. Springer, (2004), Berlin, Heidelberg. doi: 10.1007/978-3-540-28646-2_29

[30] Xu, Q., Mao, J., Jin, Z., Simulated annealing-based ant colony algorithm for tugboat scheduling optimization. *Mathematical Problems in Engineering*, (2012). doi: 10.1155/2012/246978.

[31] Kaveh, A., Hassani, B., Shojaee, S., & Tavakkoli, S. M. Structural topology optimization using ANT colony methodology. *Engineering Structures*, *30*(9), (2008), 2559–2565. doi: 10.1016/j.engstruct.2008.02.012

[32] Luh, G.-C., & Lin, C.-Y. Structural topology optimization using ant colony optimization algorithm. *Applied Soft Computing*, *9*(4), (2009), 1343–1353. doi: 10.1016/j.asoc.2009.06.001

[33] Kaveh, A., Talatahari, S., An improved ant colony optimization for the design of planar steel frames. *Engineering Structures*, 32, (2010), 864-873. doi: 10.1016/j.engstruct.2009.12012

[34] Angelo, J.S., Bernardino, H.S., Barbosa, H.J.C., Multi-objective ant colony approaches for structural optimization problems.Proceedings of the Eleventh International Conference on Computational Structures Technology, Stirlingshire, Scotland, 2012.

[35] Wang, D., Tan, D., & Liu, L. Particle Swarm Optimization Algorithm: An overview. *Soft Computing*, *22*(2), (2018), 387–408. doi: 10.1007/s00500-016-2474-6

[36] Jain, N.K., Nangia, U., Jain, J., A Review of particle swarm optimization. *J.Inst.Eng.India Ser. B*, 99(4), (2018), 407-411. doi: 10.1007/s40031-018-0323-y

[37] Zeng, N., Qiu, H., Wang, Z., Liu, W., Zhang, H., & Li, Y. A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of alzheimer's disease. *Neurocomputing*, *320*, (2018), 195–202. doi: 10.1016/j.neucom.2018.09.001

[38] Jain, I., Jain, V. K., & Jain, R. Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. *Applied Soft Computing*, *62*, (2018), 203–215. doi: 10.1016/j.asoc.2017.09.038

[39] Zarei, A., Mousavi, SF., Eshaghi Gordji, M. *et al.* Optimal Reservoir Operation Using Bat and Particle Swarm Algorithm and Game Theory Based on Optimal Water Allocation

among Consumers. *Water Resour Manage* 33, (2019), 3071–3093. doi: 10.1007/s11269-019-02286-9

[40] Cao, Y., Ye, Y., Zhao, H., Jiang, Y., Wang, H., Shang, Y., & Wang, J. Remote Sensing of water quality based on HJ-1A HSI imagery with modified discrete binary particle swarm optimization-partial least squares (MDBPSO-PLS) in inland waters: A case in Weishan Lake. *Ecological Informatics*, *44*, (2018), 21–32. doi: 10.1016/j.ecoinf.2018.01.004

[41] Mohebbi, A., Achiche, S., & Baron, L. Integrated and concurrent detailed design of a mechatronic quadrotor system using a fuzzy-based particle swarm optimization. *Engineering Applications of Artificial Intelligence*, *82*, (2019), 192–206. doi: 10.1016/j.engappai.2019.03.025

[42] Song, M., Chen, K., & Wang, J. Three-dimensional wind turbine positioning using gaussian particle swarm optimization with differential evolution. *Journal of Wind Engineering and Industrial Aerodynamics*, *172*, (2018), 317–324. doi: 10.1016/j.jweia.2017.10.032

[43] Shen, J., Han, L. RETRACTED ARTICLE: Design process optimization and profit calculation module development simulation analysis of financial accounting information system based on particle swarm optimization (PSO). *Inf Syst E-Bus Manage* 18, (2020), 809–822. doi: 10.1007/s10257-018-00398-0

[44] Yi, T., Zheng, H., Tian, Y., & Liu, J. Intelligent prediction of transmission line project cost based on least squares support vector machine optimized by particle swarm optimization. *Mathematical Problems in Engineering*, (2018), 1–11. doi: 10.1155/2018/5458696

[45] Luh, G.-C., Lin, C.-Y., & Lin, Y.-S. A binary particle swarm optimization for continuum structural topology optimization. *Applied Soft Computing*, *11*(2), (2011), 2833–2844. doi: 10.1016/j.asoc.2010.11.013

[46] Mortazavi, A., Toğan, V., Simuitaneous size, shape and topology optimization of truss structures using integrated particle swarm optimizer. *Struct Multidisc Optim*, 54, (2016), 715-736. doi: 10.1007/s00158-016-1449-7

[47] Dongare, A. D., Kharde, R. R., & Kachare, A. D. Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology (IJEIT)*, *2*(1), (2012), 189-194.

[48] Abdolrasol, M.G.M., Hussain, S.M.S., Ustun, T.S., Sarker, M.R., Hannan, M.A., Mohamed, R., Ali, J.A., Mekhilef, S., Milad, A., Artificial neural networks based optimization techniques: a review. *Electronics*, 10, (2021), 2689. doi: 10.3390/electronics10212689

[49] Lee Sian Choon, Samsudin, A., & Budiarto, R. (n.d.). Lightweight and cost-effective MPEG video encryption. *Proceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications,* 2004, 525–526. doi: 10.1109/ictta.2004.1307863

[50] D. Cho, Y. -W. Tai and I. S. Kweon, "Deep Convolutional Neural Network for Natural Image Matting Using Initial Alpha Mattes," in *IEEE Transactions on Image Processing*, vol. 28, no. 3, March 2019, pp. 1054-1067. doi: 10.1109/TIP.2018.2872925.

[51] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* 521, (2015), 436–444. doi: 10.1038/nature14539

[52] Himmelblau, D.M., Applications of artificial neural networks in chemical engineering. *Korean J. Chem. Eng.*, 17(4), (2000), 373-392.

[53] Adeli, H., Neural networks in civil engineering 1989-2000. Computer-aided civil and infrastructure engineering, 16, (2001), 126-142.

[54] Banga, Saurabh & Gehani, Harsh & Bhilare, Sanket & Patel, Sagar & Kara, Levent. (2018). 3D Topology Optimization using Convolutional Neural Networks. 10.48550/arXiv.1808.07440.

[55] Meng, Z., Lv, S., Gao, Y., Zhong, C., & An, K. Data-driven reliability-based topology optimization by using the extended multi scale finite element method and neural network approach. *Computer Methods in Applied Mechanics and Engineering*, *438*, (2025), 117837. doi: 10.1016/j.cma.2025.117837

[56] Chandrasekhar, A., Suresh, K. TOuNN: Topology Optimization using Neural Networks. *Struct Multidisc Optim* 63, (2021), 1135–1149. doi: 10.1007/s00158-020-02748-4

[57] Xing, Y., & Tong, L. A stochastic gradient online learning and prediction method for accelerating structural topology optimization using recurrent neural network. *Engineering Structures*, *338*, (2025), 120507. doi: 10.1016/j.engstruct.2025.120507

[58] Nayak, J., Swapnarekha, H., Naik, B., Dhiman, G., Vimal, S. 25 years of particle swarm optimization: flourishing voyage of two decades. *Archives of Computational Methods in Engineering*, 30, (2023), 1663-1725. doi: 10.1007/s11831-022-09849-x.

[59] Alhijawi, B., Awajan, A. Genetic algorithms: theory, genetic operators, solutions, and applications. Evolutionary Intelligence, 17, (2024), 1245-1256. doi: 10.1007/s12065-023-00822-6

[60] M. R. G. Meireles, P. E. M. Almeida and M. G. Simoes, A comprehensive review for industrial applicability of artificial neural networks, in *IEEE Transactions on Industrial Electronics*, 50(3), (2003), 585-601. doi: 10.1109/TIE.2003.812470.