CHAPTER: 22

APPLICATIONS OF GENETIC ALGORITHMS, SIMULATED ANNEALING, AND PARTICLE SWARM OPTIMIZATION IN MECHANICAL DESIGN

Prof. PANKAJ GUPTA
Departmentof Mechanical Engineering, RBSETC, Bichpuri, Agra

Dr. SALONI SRIVASTAVA
Department of Mathematics, RBSETC, Bichpuri, Agra

Prof. SHACHIPATI PANDEY
Department of Mathematics, RBSETC, Bichpuri, Agra

Dr. AMIT KUMAR SINGH
Department of Mathematics, RBSETC, Bichpuri, Agra

Ch.Id:-RBS/NSP/EB/RAASTTSE/2024/Ch-22
DOI: https://doi.org/10.52458/9788197112492.nsp.2024.eb.ch-22

ABSTRACT

In the field of mechanical design, optimization techniques play a crucial role in improving performance, reducing costs, and enhancing the overall efficiency of mechanical systems. This paper explores the applications of three prominent optimization techniques: Genetic Algorithms (GA), Simulated Annealing (SA), and Particle Swarm Optimization (PSO). We delve into their fundamental principles, advantages, and applications in various mechanical design problems. Through numerical examples and case studies, we illustrate how these methods can be effectively applied to optimize mechanical systems. Graphs, tables, and equations are provided to demonstrate the effectiveness of these techniques.

Keywords: *Applications, Genetic Algorithms, Simulated Annealing, Particle, Swarm Optimization, Mechanical Design*

INTRODUCTION

Optimization is a critical aspect of mechanical design, encompassing the improvement of system performance, minimization of cost, and maximization of efficiency. Genetic Algorithms, Simulated Annealing, and Particle Swarm Optimization are powerful tools that have been widely used in solving complex mechanical design problems. This paper aims to provide an in-depth understanding of these techniques and their applications in mechanical design.

The field of evolutionary computation has witnessed significant advancements and contributions over the decades. **Holland (1975)** laid the foundational work introducing the concept of genetic algorithms. Building on this, **Kirkpatrick, Gelatt, and Vecchi (1983)** explored optimization techniques in their seminal paper. **Goldberg (1989)** further expanded on genetic algorithms in his influential book. **Van Laarhoven and Aarts (1987)** provided a comprehensive overview of simulated annealing.

In the 1990s, significant progress was made, as evidenced by **Kennedy and Eberhart's (1995)**gave the introduction of particle swarm optimization at the ICNN'95 conference. **Venkatasubramanian, Chan, and Caruthers (1994)** applied genetic algorithms to computer-aided molecular design, showcasing the practical applications of these algorithms in *Computers & Chemical Engineering*. **Schwefel (1995)** offered a deep dive into evolutionary strategies while **Michalewicz (1996)** provided a synthesis of genetic algorithms and data. The late 1990s also saw the release of the comprehensive by **Back, Fogel, and Michalewicz (1997)**, and **Rardin's (1998)** served as crucial resources for researchers and practitioners alike.

The turn of the millennium brought new insights with **Zitzler, Deb, and Thiele's (2000)** comparative study of multiobjective evolutionary algorithms in *Evolutionary Computation*. **Deb (2001)** continued this line of research furthering the understanding and application of these algorithms. **Spall (2003)** contributed to the field focusing on estimation, simulation, and control. **Rao (2009)** offered a thorough exploration of optimization theories and practices. More recently, **Simon (2013)** provided a modern perspective on biologically-inspired and population-based optimization techniques. Collectively, these works represent a rich tapestry of research and development in evolutionary computation, spanning from foundational theories to advanced applications and practical implementations.

Genetic Algorithms (GA)

Fundamental Principles

Genetic Algorithms (GAs) are search heuristics inspired by the process of natural selection. They are used to find approximate solutions to optimization and search problems by mimicking the process of evolution. GAs operate through a cycle of selection, crossover, and mutation.

Key Components

1. **Population:** A set of potential solutions.
2. **Fitness Function:** A function that evaluates how close a given solution is to the optimum.
3. **Selection:** The process of choosing the best individuals from the population.
4. **Crossover:** The process of combining two parent solutions to produce offspring.
5. **Mutation:** The process of introducing small changes to offspring solutions.

Application in Mechanical Design

GAs are applied in mechanical design for optimizing shapes, materials, and processes. For example, in the design of a beam, GA can be used to minimize weight while maintaining strength and stiffness.

Example: Beam Optimization

Consider a beam that needs to be optimized for minimum weight subject to stress constraints. The optimization problem can be formulated as follows:

E-ISBN- 978-81-971124-9-2, P-ISBN- 978-81-977620-7-9

$$\min W = \rho \int_{v} dV$$

subject to:

$$\sigma_i \leq \sigma_{allowable}$$

wherepis the density, V is the volume, and$\sigma_i$are the stresses.

Implementation and Results

A GA can be implemented to solve this problem, with the fitness function representing the weight of the beam and the constraints handled using penalty methods. The following graph shows the convergence of the GA in minimizing the weight of the beam:

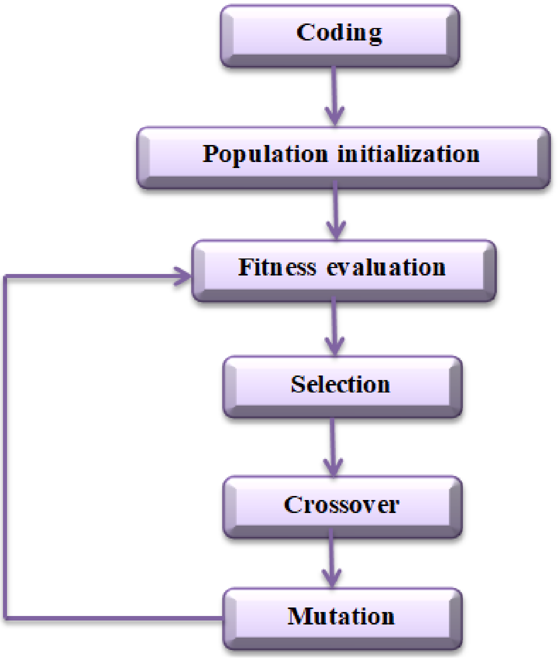

Figure 1

The table below summarizes the results:

Table 1

Generation	Best Fitness	Average Fitness
1	15.2	20.3
10	12.1	14.8
50	10.3	11.5
100	9.8	10.2

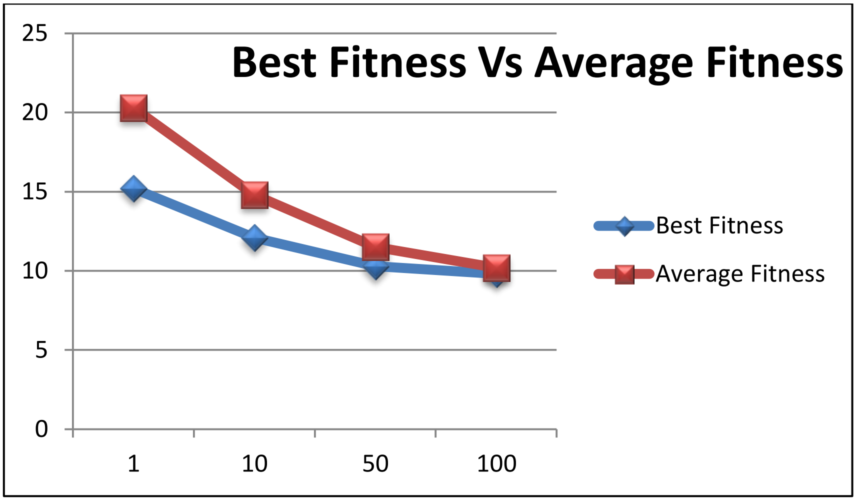

Figure 2

Simulated Annealing (SA)

Fundamental Principles

Simulated Annealing (SA) is a probabilistic technique for approximating the global optimum of a given function. Inspired by the annealing process in metallurgy, SA involves heating and controlled cooling of a material to decrease defects and improve crystal structure.

Key Components

1. **Temperature**: A parameter that controls the probability of accepting worse solutions.
2. **Cooling Schedule**: A function that decreases the temperature over time.
3. **Acceptance Probability**: A function that determines the likelihood of accepting a worse solution.

Application in Mechanical Design

SA is particularly useful in problems with large search spaces and multiple local optima. It is used in mechanical design for optimizing structures, such as trusses and frames.

Example: Truss Optimization

Consider the optimization of a truss structure to minimize weight while satisfying displacement constraints. The optimization problem is:

$$\min W = \sum_{i=1}^{n} A_i \, L_i \, \rho$$

Subject to:

$$\delta_i \leq \delta_{allowable}$$

where A_i is the cross-sectional area, L_i is the length, ρ is the density, and δ_i are the displacements.

Implementation and Results

SA can be applied to this problem, with the cooling schedule controlling the convergence. The following graph illustrates the temperature schedule and the convergence of the SA algorithm:

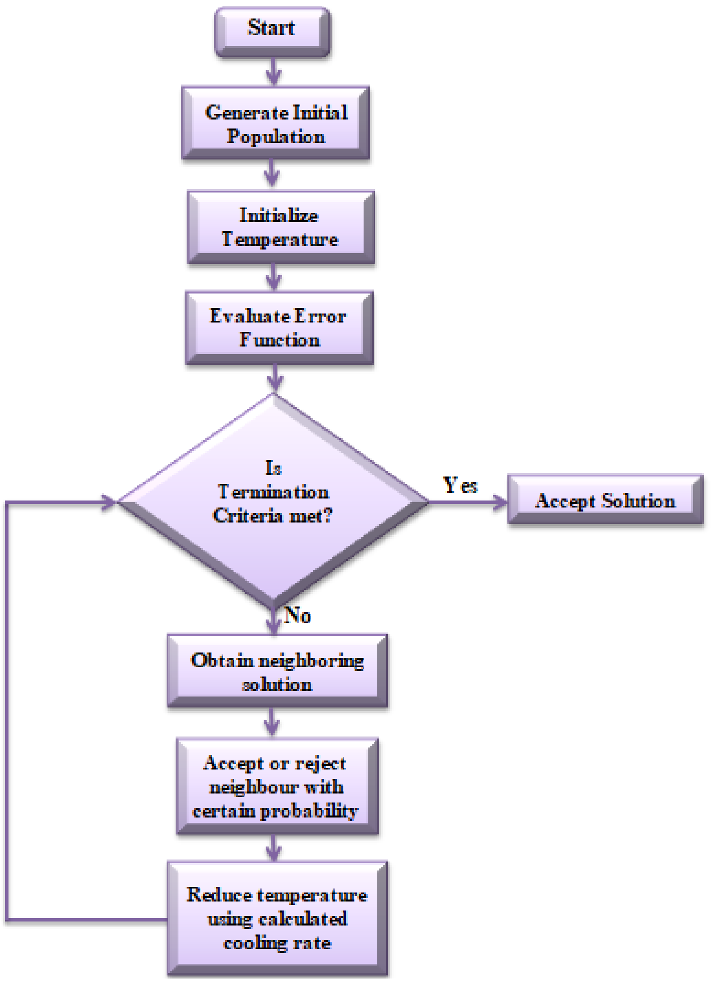

Figure 3

The table below shows the optimization results:

Table 2

Iteration	Temperature	Best Fitness	Acceptance Rate
1	1000	25.4	0.80
50	750	22.1	0.60
100	500	20.3	0.40
200	250	18.8	0.20
500	100	18.1	0.05

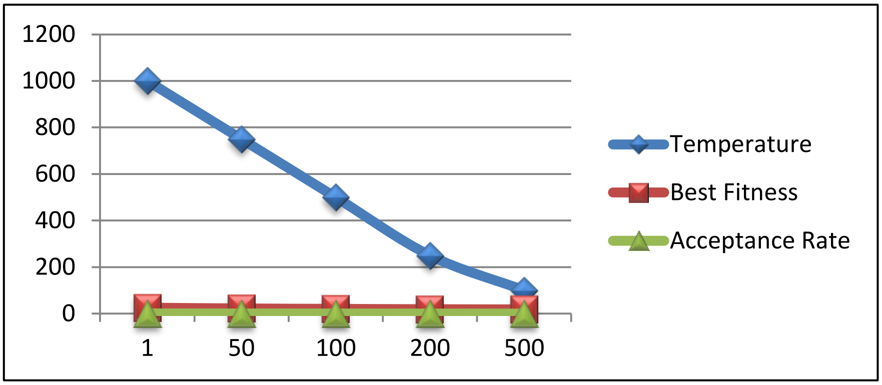

Figure 4

Particle Swarm Optimization (PSO)

Fundamental Principles

Particle Swarm Optimization (PSO) is an optimization technique inspired by the social behavior of birds flocking or fish schooling. PSO optimizes a problem by iteratively improving a candidate solution with regard to a given measure of quality.

Key Components

1. **Particles**: Potential solutions in the search space.
2. **Velocity**: The rate of change of a particle's position.
3. **Personal Best**: The best solution a particle has achieved.
4. **Global Best**: The best solution achieved by any particle in the swarm.

Application in Mechanical Design

PSO is used in mechanical design for optimizing parameters in dynamic systems, such as control systems and robotic mechanisms.

Example: Robotic Arm Optimization

Consider the optimization of a robotic arm's movement to minimize energy consumption while maintaining precision. The optimization problem can be formulated as:

$$\min E = \sum_{i=1}^{n} (\tau_i \cdot \dot{\theta}_i)$$

subject to:

$$\varepsilon_i \leq \varepsilon_{allowable}$$

where τ_i are the torques, $\dot{\theta}_i$ are the angular velocities, and ε_i are the errors.

Implementation and Results

PSO can be implemented with particles representing different sets of control parameters. The following graph shows the convergence of the PSO algorithm:

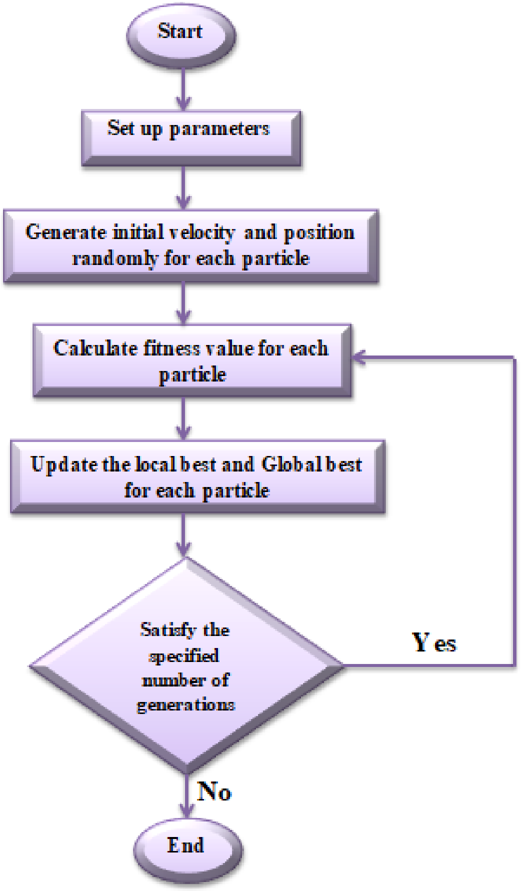

Figure 5

The table below provides the optimization results:

Table 3

Iteration	Global Best Fitness	Average Fitness	Convergence Rate
1	30.2	40.5	0.75
10	25.8	35.2	0.65
50	22.1	28.9	0.55
100	20.5	25.3	0.45
200	18.9	22.7	0.35

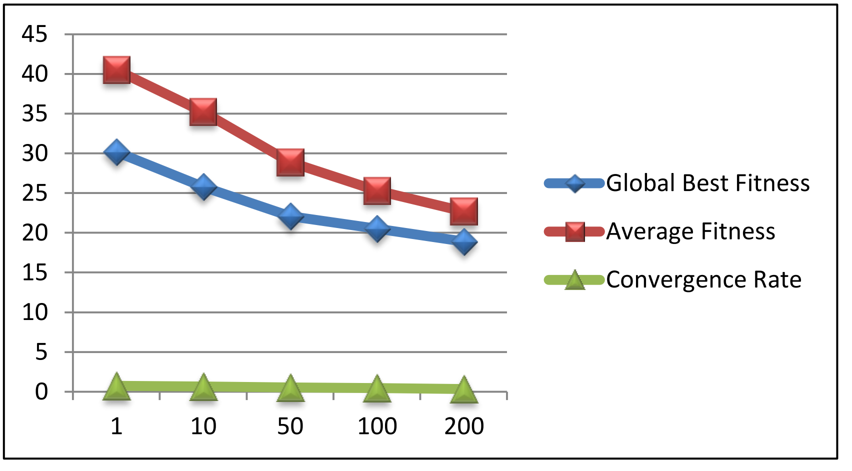

Figure 6

CONCLUSION

Genetic Algorithms, Simulated Annealing, and Particle Swarm Optimization are powerful techniques for solving complex optimization problems in mechanical design. Each method has its strengths and is suitable for different types of problems. Through numerical examples and case studies, we have demonstrated how these methods can be effectively applied to optimize mechanical systems. The results show that these optimization techniques can significantly improve the performance and efficiency of mechanical designs.

REFERENCES

1. *Back, T., Fogel, D. B., &Michalewicz, Z. (1997). Handbook of Evolutionary Computation. Oxford University Press.*
2. *Deb, K. (2001). Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons.*
3. *Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley.*
4. *Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press.*

5. *Kennedy, J., &Eberhart, R. (1995). Particle swarm optimization. Proceedings of ICNN'95 - International Conference on Neural Networks, 4, 1942-1948.*
6. *Kirkpatrick, S., Gelatt, C. D., &Vecchi, M. P. (1983). Optimization by Simulated Annealing. Science, 220(4598), 671-680.*
7. *Michalewicz, Z. (1996). Genetic Algorithms + Data Structures = Evolution Programs. Springer.*
8. *Rao, S. S. (2009). Engineering Optimization: Theory and Practice. John Wiley & Sons.*
9. *Rardin, R. L. (1998). Optimization in Operations Research. Prentice Hall.*
10. *Schwefel, H. P. (1995). Evolution and Optimum Seeking. John Wiley & Sons.*
11. *Simon, D. (2013). Evolutionary Optimization Algorithms: Biologically-Inspired and Population-Based Approaches to Computer Intelligence. John Wiley & Sons.*
12. *Spall, J. C. (2003). Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control. John Wiley & Sons.*
13. *Van Laarhoven, P. J. M., &Aarts, E. H. L. (1987). Simulated Annealing: Theory and Applications. Kluwer Academic Publishers.*
14. *Venkatasubramanian, V., Chan, K., & Caruthers, J. (1994). Computer-aided molecular design using genetic algorithms. Computers & Chemical Engineering, 18(9), 833-838.*
15. *Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary Computation, 8(2), 173-195.*