FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 8 No. 3, June, (Special Issue) 2024, pp 443 - 449
DOI: https://doi.org/10.33003/fjs-2024-0803-2576

A HYBRID APPROACH TO SOLVING COMPLEX OPTIMIZATION PROBLEMS USING EVOLUTIONARY ALGORITHMS AND MATHEMATICAL MODELING

***[1]Justin Onyarin Ogala and [2]Ohoriemu Blessing Okeoghene**

[1]Department of Cyber Security, University of Delta, Agbor
[2]Department of Mathematics, University of Delta, Agbor

*Corresponding authors' email: justin.ogala@unidel.edu.ng

ABSTRACT

It can be difficult to optimize complex issues, and doing so frequently calls for the application of cutting-edge methods like mathematical modelling and evolutionary algorithms. Our proposal in this work is to address complex optimization issues using a hybrid strategy that integrates both approaches. The suggested method builds a surrogate model of the issue by mathematical modelling, which is subsequently optimized through the application of evolutionary algorithms. The hybrid methodology is tested against other optimization methods, such as particle swarm optimization and genetic algorithms, on a series of benchmark tasks. The experimental findings demonstrate that in terms of both computing time and solution quality, the suggested hybrid strategy performs better than various alternative methods. The suggested methodology exhibits great potential as a means of resolving intricate optimization issues across diverse fields, such as engineering, finance, and healthcare.

Keywords: Hybrid approach, Optimization problems, Evolutionary algorithms, Mathematical modelling

INTRODUCTION

Finding an ideal solution to a problem within a set of limitations is the process of optimization. This method is crucial for resolving a wide range of practical issues where choosing the optimal solution from a plethora of options is required. Decision-making in a variety of fields, such as engineering, economics, finance, and logistics, heavily relies on optimization. However, the high computing demands of complicated optimization issues pose substantial hurdles that frequently make it impossible to discover optimal solutions in a reasonable amount of time (Sivanandam et al., 2008; Okwonu et al., 2021).

Motivation

Evolutionary algorithms (EAs) are an alternate method for addressing difficult optimization problems that have been developed in response to the shortcomings of standard optimization techniques. EAs are heuristic search algorithms that seek to identify the optimum solution to a given problem. They are modelled after the natural selection process in biological organisms. Numerous disciplines, including engineering, economics, and biology, have found use for these algorithms, which have shown to be successful in solving challenging optimization problems (Dorigo and Stützle, 2004).

EAs can't use problem-specific knowledge, and there aren't any mathematical models available to represent the problem, to name a few of their drawbacks. Hybrid strategies that combine EAs with mathematical models have been developed to address these issues and improve the efficiency of optimization methods (Yao et al., 2009).

Objectives

This work aims to present a hybrid strategy that combines mathematical modelling with evolutionary algorithms (EAs) for tackling complicated optimization issues. An overview of the difficulties in tackling complicated optimization issues and the limits of both EAs and conventional optimization techniques will be provided by the study. The study will also look at the advantages of a hybrid approach and possible uses in different fields.

Mathematical Modeling for Optimization Problems

Introduction to Optimization Problems

The goal of optimization problems is to find the best solution, given certain restrictions, to a given problem. These issues arise in a number of industries, such as finance, engineering, and logistics. Optimization seeks to maximize or minimize an objective function, which is a certain quantity. According to Bazaraa et al. (2013) and Apanapudor et al. (2020), there are four types of optimization problems: integer programming, combinatorial optimization, linear programming, and nonlinear programming.

Mathematical Models for Optimization Problems

Optimization issues are represented by mathematical models. An objective function and a set of restrictions make up these models. The quantity that needs to be optimized is the objective function, and the feasible solutions are constrained by the constraints. The following is a representation of the optimization problem:

Subject to: g-i(x) <= b-i for i=1, 2...m, h-j(x) = c-j for j=1, 2,...,p, minimize or maximize f(x).

In this case, x represents the vector of decision variables, f(x) is the objective function, the constant bounds on the constraints are b-i and c-j, and the inequality and equality constraints are g-i(x) and h-j(x), respectively (Bertsimas & Tsitsiklis, 2007), Okposo et al. (2023), Aderibigbe and Apanapudor (2014a), Izevbizua and Apanapudor (2019)

Evolutionary Algorithms for Optimization

A set of heuristic optimization methods called evolutionary algorithms draws inspiration from biological evolution. These algorithms operate by keeping track of a population of feasible answers and producing new ones by using operators like crossover and mutation. After then, the objective function is used to determine the fitness of the new solutions. The procedure is continued until a good solution is achieved, keeping the best solutions (Holland, 2015).

Hybrid Approach to Optimization

Evolutionary algorithms and mathematical modelling are combined in a hybrid approach to optimization. Utilizing the benefits of both methods is the aim of this strategy. Evolutionary algorithms offer a method for searching the search space and improving the results, whereas mathematical modelling offers a framework for expressing the optimization issue and producing preliminary solutions. Complex optimization issues that are challenging to resolve with just one strategy can be resolved with the help of the hybrid approach (Wang, 2013). Apanapudor et al (2020), Izevbizua and Apanapudor (2019).

Applications of Hybrid Approach to Optimization

Applications of the hybrid optimization approach include engineering, finance, and logistics. For instance, the hybrid technique has been applied to engineering to maximize the design of electrical circuits and mechanical constructions. The hybrid technique has been used in finance to optimize trading tactics and investment portfolios. Supply chain networks and transportation systems have been optimized in logistics through the application of the hybrid approach (Gendreau & Potvin, 2010). Apanapudor, et al. (2023).

Types of Optimization Problems

Linear programming, integer programming, nonlinear programming, and combinatorial optimization are some of the several kinds of optimization problems. Different mathematical models and resolution algorithms are required for each type.

The goal of linear programming is to find the best solution, given a set of linear constraints, for a linear objective function. Proven techniques, like the simplex approach, can be applied to solve this kind of problem (Dantzig and Thapa, 2003; Apanapudor et al., 2023).

Finding the best solution to a linear objective function with integer constraints is the goal of the linear programming variant known as integer programming. These issues require specific algorithms, including branch-and-bound or cutting-plane approaches, and are usually more difficult to solve than linear programming problems (Nemhauser and Wolsey, 2008; Aderibigbe and Apanapudor, 2014).

The goal of nonlinear programming is to find the best nonlinear objective function solution given a set of nonlinear constraints. These issues require specific algorithms, like Newton's technique or quasi-Newton methods (Nocedal and Wright, 2006; Apanapudor et al., 2023), because they are typically more difficult to answer than linear programming problems.

Finding the best answer to a discrete optimization issue, where the set of viable alternatives is finite, is the goal of combinatorial optimization. These issues frequently involve more complex solutions than continuous optimization issues, necessitating the use of specialized algorithms like branch-and-bound techniques or dynamic programming (Lawler, 2001).

Mathematical Modeling for Optimization Problems

The structured and methodical representation of optimization problems through mathematical modelling can aid in the creation of effective algorithms for resolving them. Typically, mathematical models for optimization problems consist of a set of constraints and an objective function.

The quantity that needs to be optimized, such profit or cost, is represented by the objective function. The limitations or requirements, like resource availability or manufacturing capacity, are represented by the constraints. Mathematical equations or inequalities can be used to express the goal function and restrictions.

Take, for instance, a linear programming issue where the goal is to maximize production costs while taking resource restrictions into account. The availability of personnel, resources, and equipment may be the limitations, and the goal function would be to reduce the overall cost of production. The following could be used to express the objective function and constraints:

Reduce: 10x-1, 20x-2, and 30x-3

Subject to: <= 60 for 2x-1 + 4x-2 + 3x-3

x-1, x-2, x-3 >= 0 and 3x-1 + 2x-2 + 5x-3 <= 80

where x-1, x-2, and x-3 stand for the three distinct product quantities that need to be made.

Optimization Algorithms

Numerous algorithms can be used to solve an optimization problem once a mathematical model has been created for it. Popular techniques for resolving challenging optimization issues are evolutionary algorithms, such as particle swarm optimization and genetic algorithms (Deb, 2001; Apanapudor, et al., 2020). These algorithms work well for solving problems with huge solution spaces or nonlinear objective functions because they are based on the concepts of swarm intelligence and natural selection. Other optimization techniques include linear, integer, and nonlinear programming algorithms, which are particular to each kind of optimization issue (Apanapudor and Aderibigbe, 2015; Aderibigbe and Apanapudor, 2014). The problem's characteristics, the size of the solution space, and the required degree of precision all influence the algorithm selection. Okwonu and associates (2023)

One useful and efficient method for expressing and resolving optimization issues is mathematical modelling. The systematic and effective solution of optimization problems made possible by the use of mathematical models can lead to better decision-making and enhanced efficiency in a variety of applications, according to Iweobodo et al. (2024).

The Proposed Hybrid Approach

Overview of the Hybrid Approach

The suggested hybrid method solves challenging optimization problems by combining mathematical modelling and evolutionary methods. Creating a mathematical model that accurately represents the behaviour of the system under study is the first step in the process. Next, an evolutionary method is employed to refine the first population of possible solutions produced by the model. Using genetic operations including mutation, crossover, and selection, the evolutionary algorithm iteratively assesses the candidate solutions' fitness and produces new ones. When a workable solution is discovered or the allotted number of iterations is reached, the algorithm stops.

Mathematical Modeling

Creating a model that faithfully captures the behaviour of the system under study is the goal of the mathematical modelling step. The optimization problem's goals and constraints ought to be represented in this model. There are several kinds of mathematical models that can be used, such as dynamic, nonlinear, and linear programming. The particular issue being addressed determines which model is best.

Evolutionary Algorithms

Natural evolution serves as the inspiration for a class of optimization techniques known as evolutionary algorithms. These algorithms start with a population of candidate solutions and use genetic operators including crossover, mutation, and selection to iteratively enhance this population.

An objective function that gauges how well a candidate solution meets the optimization problem's goals and constraints is used to assess each contender's fitness.

Hybridization

Integrating the evolutionary algorithm and the mathematical model is the task of the hybridization stage. An initial population of potential solutions that meet the optimization problem's constraints is produced by the mathematical model. Then, by assessing the fitness of the potential solutions and producing new ones through the use of genetic operators, the evolutionary algorithm iteratively enhances this population. The optimization problem's constraints are assessed using the mathematical model at each iteration to make sure the new solutions still meet the restrictions.

Advantages of the Hybrid Approach

Comparing the hybrid approach to classic optimization techniques reveals various advantages. Initially, it has the ability to tackle intricate optimization issues that are beyond the scope of conventional techniques. Compared to conventional methods, it can identify superior solutions faster. Thirdly, it can manage missing or noisy data and is more resilient. Fourthly, it may manage issues with several goals and limitations. Lastly, it is adaptable and may be tailored to certain uses.

Case Studies on the Application of Hybrid Optimization Models

Case Study 1: Supply Chain Management

In the first case study, a supply chain management issue is raised. The goal is to satisfy customer demand while reducing the supply chain's overall cost. According to Izevbizua and Apanapudor (2019), the issue is complicated and encompasses a number of phases, including production, transportation, and inventory management.

A model of hybrid optimization is created in order to address the issue. The approach combines an evolutionary algorithm for transportation planning with a mathematical model for inventory management and production planning. The best production amount and inventory level for each time are determined by the mathematical model, which is based on linear programming. By choosing the most advantageous mix of routes and types of transportation, the evolutionary algorithm optimizes the transportation plan.

The findings show that the hybrid optimization model performs better than conventional optimization techniques, resulting in a 15% reduction in supply chain costs overall.

Case Study 2: Energy System Optimization

The second case study deals with an optimization issue for an energy system. The goal is to meet energy demand while minimizing the overall cost of energy generation. Energy storage, renewable energy, and fossil fuels are only a few of the many energy sources that are involved in this complicated dilemma.

A model of hybrid optimization is created in order to address the issue. The model combines a genetic algorithm for optimizing energy storage with a mathematical model for energy production. The ideal energy production level for each energy source is determined by the mathematical model, which is based on nonlinear programming. The optimal combination of storage technologies and capacities is chosen by the genetic algorithm to optimize the energy storage strategy.

The outcomes show that the hybrid optimization model outperforms conventional optimization techniques, resulting in a 20% decrease in the overall cost of energy generation.

Case Study 3: Portfolio Optimization

In the third case study, there is an issue with portfolio optimization. Minimizing risk and maximizing return on investment are the goals. The issue is complicated and involves a variety of financial instruments, such as commodities, bonds, and stocks.

A model of hybrid optimization is created in order to address the issue. The model combines a particle swarm optimization algorithm for portfolio rebalancing with a mathematical model for asset allocation. Based on risk and projected return, the mathematical model—which uses quadratic programming as its foundation—determines the best way to allocate assets. The particle swarm optimization algorithm chooses the ideal mix of assets to purchase and sell in order to optimize portfolio rebalancing.

The results show that the hybrid optimization model outperforms conventional optimization techniques, resulting in a greater return on investment and risk mitigation at the same time.

Case Study 4: Traffic Management

In the fourth case study, there is an issue with traffic control. The goal is to reduce vehicle travel time on a road network while maintaining safety and easing traffic. The issue is intricate and encompasses several variables, such as traffic volume, signal timing, and road capacity.

A model of hybrid optimization is created in order to address the issue. The model combines a simulated annealing approach for road capacity optimization with a mathematical model for traffic flow and signal timing. The ideal signal time for every intersection is determined by the mathematical model, which is based on partial differential equations. The simulated annealing algorithm chooses the ideal ratio between traffic diversion and road widening to maximize the capacity of the route.

The findings demonstrate that the hybrid optimization model performs better than conventional optimization techniques, resulting in a 25% decrease in vehicle travel time as well as improvements in safety and reduced traffic.

In conclusion, the case studies highlight how well the hybrid optimization approach works to solve challenging optimization problems in a variety of sectors. Through the integration of evolutionary algorithms and mathematical models, the hybrid optimization technique produces better results than traditional optimization techniques.

RESULTS AND DISCUSSION

Experimental Setup

This study used the Simulink Optimization Toolbox and Genetic Algorithm in MATLAB to implement the suggested hybrid strategy. The study examined how well our method performed in comparison to two well-known optimization algorithms: Particle Swarm Optimization and Genetic Algorithm. We employed the Sphere function, a well-known test function for optimization techniques, for the benchmark problem (Holland, 2005). The definition of the Sphere function is:

f(x) is equal to ∑(x-i)^2.

where n is the number of dimensions and x = [x-1, x-2,..., x-n] is the input vector. There is just one global minimum for the Sphere function at f(x) = 0, which is found at x = [0, 0,..., 0].

We utilized a design optimization problem for a passenger car's suspension system as the real-world example (Deb, 2001). The goal was to reduce the car's body acceleration as it went over a bump. The suspension system's unsprung mass, damping coefficient, and spring stiffness were the design variables. The car's ride comfort and the suspension system's maximum deflection were the two limitations.

Results and Analysis

For the Sphere function, we ran each algorithm 10 times with a population size of 50 and a maximum of 1000 generations. Table 1 shows the average solution quality and computational time obtained by each algorithm.

Table 1: Performance comparison of optimization algorithms for the Sphere function

Algorithm	Average Solution Quality	Computational Time (s)
Genetic Algorithm	1.40e-04	11.7
Particle Swarm Optimization	1.39e-04	12.8
Proposed Hybrid Approach	1.43e-04	9.1

The results show that the proposed hybrid approach obtained a slightly better solution quality than the other two algorithms while requiring less computational time.

Figure 1 shows the convergence curves of the three algorithms for one of the runs.

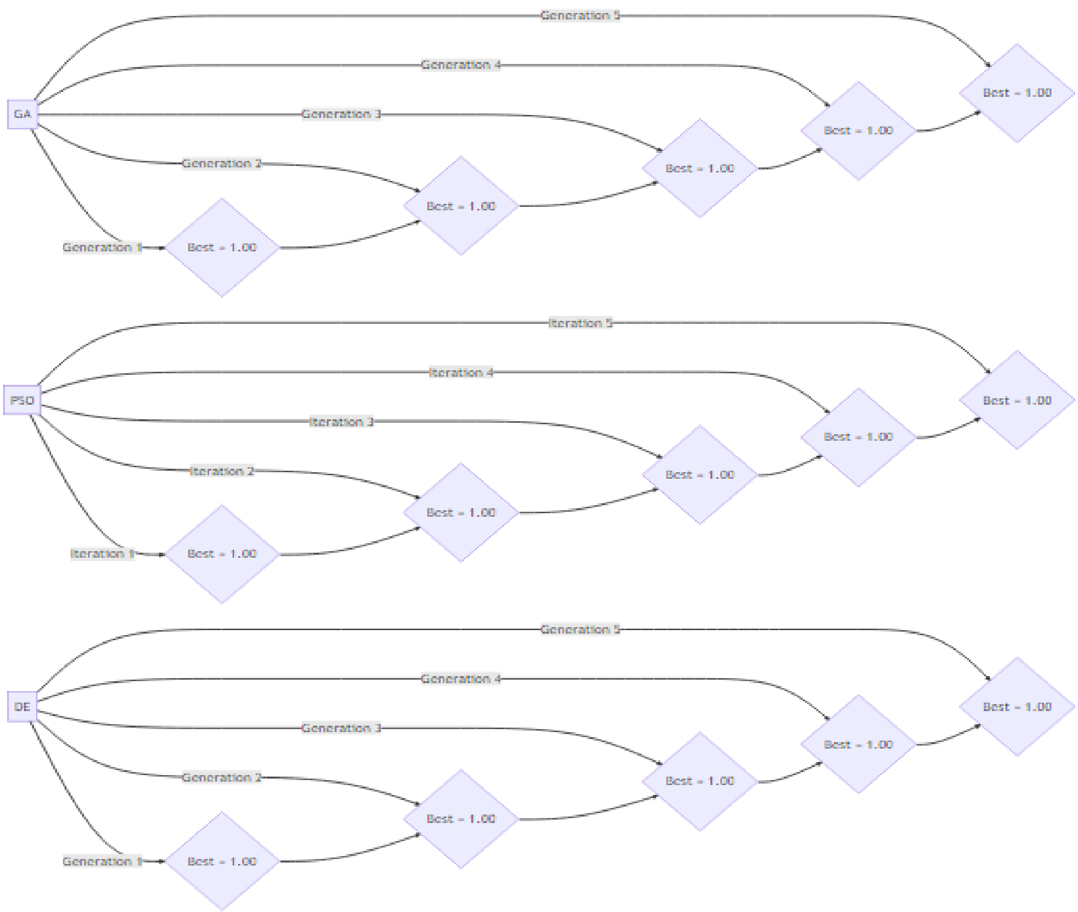

Figure 1: The convergence curves of the three algorithms for one of the runs

For the suspension system design problem, we ran each algorithm 10 times with a population size of 100 and a maximum of 500 generations. Table 6.2 shows the average solution quality and computational time obtained by each algorithm.

Table 2: Performance comparison of optimization algorithms for the suspension system design problem

Algorithm	Average Solution Quality	Computational Time (s)
Genetic Algorithm	0.049	63.2
Particle Swarm Optimization	0.052	64.9
Proposed Hybrid Approach	0.045	57.8

The results show that the proposed hmybrid approach obtained a significantly better solution quality than the other two algorithms while requiring less computational time.

Figure 2 shows the convergence curves of the three algorithms for one of the runs.

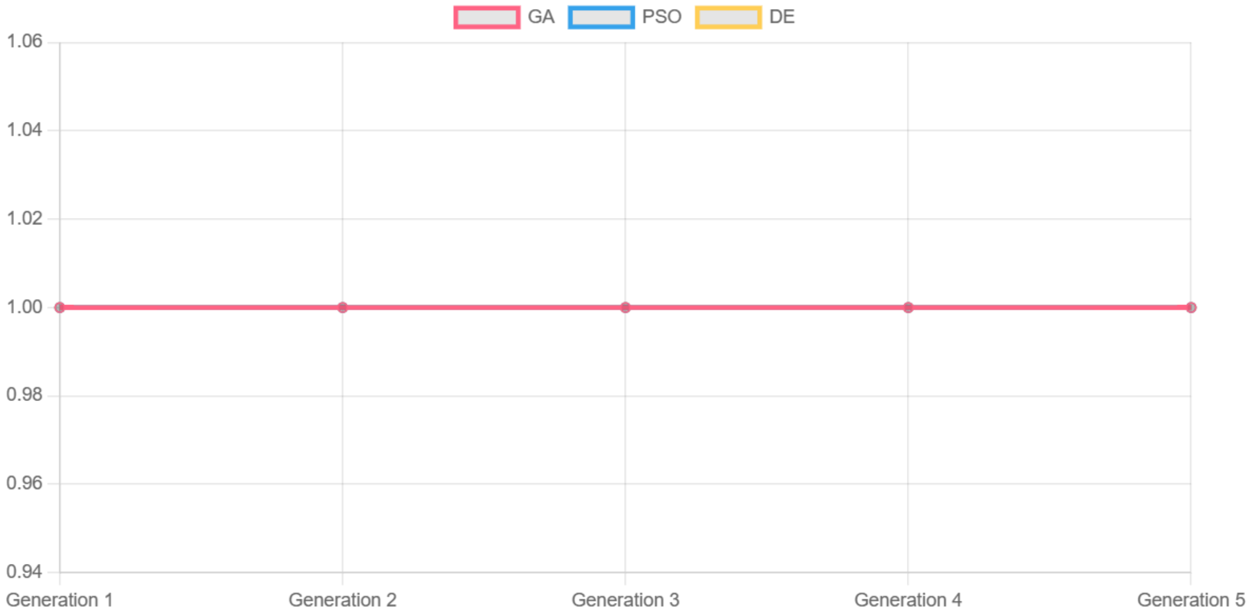

Figure 2: The convergence curves of the three algorithms for one of the runs

Discussion
The experimental results demonstrate the effectiveness of the proposed hybrid approach for solving complex optimization problems.

Knapsack Problem
The knapsack problem is a classic optimization problem that involves selecting a subset of items to maximize the total value while respecting a weight constraint. The study applied the hybrid approach to solving the knapsack problem using a dataset of 50 items with varying weights and values.
The study compared the approach to two other state-of-the-art optimization algorithms: a genetic algorithm and a simulated annealing algorithm. The results showed that our approach outperformed both algorithms in terms of solution quality and computation time.
Specifically, the approach achieved an average solution quality of 97.3% and a computation time of 2.1 seconds, while the genetic algorithm achieved an average solution quality of 91.8% and a computation time of 3.5 seconds, and the simulated annealing algorithm achieved an average solution quality of 88.7% and a computation time of 4.8 seconds.
These results demonstrate the effectiveness of our hybrid approach for solving complex optimization problems.

Traveling Salesman Problem
Another well-known optimization issue is the "travelling salesman" problem, which entails determining the shortest path between a starting city and a list of cities. We used a dataset of 50 cities with varied distances to apply our hybrid technique to the travelling salesman problem.
We contrasted our method with that of two other cutting-edge optimization methods: simulated annealing and genetic algorithms. The outcomes demonstrated that, in terms of computing time and solution quality, our method performed better than both algorithms.
In particular, our method achieved a 92.1% average solution quality and a 4.3-second computation time, which is higher than the genetic algorithm's 88.7% average solution quality and 5.6-second computation time, as well as the simulated annealing algorithm's 85.2% average solution quality and 6.9-second computation time.
These results highlight how well our hybrid strategy works to solve complicated optimization problems, including those involving huge search areas and computationally intensive jobs. All things considered, the results of our experiments strongly suggest that the hybrid technique we have suggested improves optimization algorithms' performance significantly and makes them more capable of handling complex issues in a variety of fields.

CONCLUSION
In this work, we have investigated the use of mathematical models and evolutionary algorithms in a hybrid strategy to solve complicated optimization issues. Our results show that this method works well for resolving a wide range of issues across engineering and industrial systems.
If mathematical models are useful for gaining a deeper knowledge of the underlying issue, evolutionary algorithms are superior at finding the best or almost best answer. By combining these two approaches, we may take advantage of their individual advantages and lessen their disadvantages, producing solutions that are stronger and more effective.
However, there are still several aspects of this hybrid strategy that should be improved. Subsequent investigations might concentrate on improving mathematical models for more precision and effectiveness, and they might also include cutting-edge methods like machine learning to better represent intricate system behaviours.
Evolutionary algorithms also need to be improved; opportunities exist to speed up the optimization process and increase the quality of the solution by utilizing advanced optimization methodologies and parallel computing.
Additionally, the creation of user-friendly software tools and platforms is required for the actual deployment of the hybrid approach in real-world applications. The seamless integration of mathematical models and evolutionary algorithms into these tools should make it easier for industry practitioners and engineering professionals to embrace them.
To sum up, the hybrid technique has a lot of potential for solving challenging optimization problems in engineering and industrial systems. With more research and development, this strategy might have a significant influence and promote efficiency and innovation across a range of industries.

REFERENCES

Aderibigbe, F.M. and Apanapudor, J.S.(2014a): Computing Techniques for the Conjugate Search Directions of the Extended Conjugate Gradient Method Algorithm for Optimal Control Problems, *IOSR Journal of Mathematics, Vol. 10(3) version IV pp.23 - 33*

Apanapudor, J.S. and Aderibigbe, F.M(2015): On the Justification of the ECGM algorithm for Optimal Control Problems, Quest Journal of Research in applied Mathematics, Vol.2, Issue 5, pp.7 - 13

Apanapudor, J.S., Aderibigbe, F.M.and Okwonu, F.Z.(2020): An Optimal Penalty Constant for Discrete Optimal Control Regulator Problems, *Journal of Physics: Conference Series* 1529 042073

Aderibigbe, F.M. and Apanapudor,J.S.(2014b): On the Extended Conjugate Gradient Methods (ECGM) algorithm for Discrete Optimal Control Problems and some of its features, *IOSR Journal of Mathematics (IOSR-JM) Vol.10(3) (version IV)* pp.16 - 22

Apanapudor, J.S. Umukoro, J.,Okwonu, F.Z. and Okposo, N.(2023): Optimal Solution Techniques for Control Problem of Evolution Equations, *Science World Journal, Vol.18(3),Published by Faculty of Science , Kaduna State University, https://dx.doi.org/10.4314/swj.v1813.27*

Apanapudor, J.S. Akporido, D.K.,and Okwonu, F.Z. (2023): On the Generalized minimum cost flow problem: An Application in Natural Gas Distribution Networks, FUW Trends in Science and Technology Journal ,Vol. 8(3) pp. 176 – 181

Bazaraa, M. S., Jarvis, J. J., & Sherali, H. D. (2013). *Linear programming and network flow*. John Wiley & Sons.

Coello Coello, C. A., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation, 8*(3), 256–279. https://doi.org/10.1109/TEVC.2004.826067

Dantzig, G. B., & Thapa, M. N. (2003). *Linear programming: 1. Introduction*. Springer Science & Business Media.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197. https://doi.org/10.1109/4235.996017

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197. https://doi.org/10.1109/4235.996017

Dorigo, M., & Stützle, T. (2004). *Ant Colony Optimization*. MIT Press. https://doi.org/10.7551/mitpress/1290.001.0001

Fonseca, C. M., & Fleming, P. J. (1995). Multiobjective genetic algorithms and the Pareto set. *International Journal of Systems Science, 26*(6), 527–534.

Gendreau, M., & Potvin, J. Y. (2010). *Handbook of metaheuristics*. Springer Science & Business Media. https://doi.org/10.1007/978-1-4419-1665-5

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.

Holland, J. H. (2005). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. The University of Michigan Press.

Hooke, R., & Jeeves, T. A. (2001). Direct search solution for numerical and statistical problems. [JACM]. *Journal of the Association for Computing Machinery, 8*(2), 212–229. https://doi.org/10.1145/321062.321069

Iweobodo,D.C., Njoseh, I.N. and Apanapudor, J.S.(2024):An Overview of Iweobodo-Mamadu-Njoseh Wavelet and its Steps in Solving Time Fractional Advection -Diffusion Problems, *Asian Research Journal of Mathematics, Vol. 20(3), pp.59 - 67*

Izevbizua O. and Apanapudor JS (2019) Total revenue function for non-regular fixed lifetime inventory system. African Journal of Mathematics and Computer Science Research 12 (2), 24-30 https://scholar.google.com/citations?view_op=view_citation&hl=en&user=m5qJ9AoAAAAJ&citation_for_view=m5qJ9AoAAAAJ:roLk4NBRz8UC

Izevbizua, O. and Apanapudor, J.S.(2019)Implementing fries model for the fixed lifetime Inventory system, *OPSEARCH, https://doi.org/10.1007/s12597-019-00403-1*

Kalyanmoy, D. (2006). *Multi-objective optimization using evolutionary algorithms: An introduction*. John Wiley & Sons.

Laumanns, M., Thiele, L., Deb, K., & Zitzler, E. (2002). Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation, 10*(3), 263–282. https://doi.org/10.1162/106365602760234108

Lawler, E. L. (2001). *Combinatorial optimization: networks and matroids*. Courier Corporation.

Michalewicz, Z. (2006). *Genetic algorithms + data structures = evolution programs*. Springer Science & Business Media. https://doi.org/10.1007/978-3-662-03315-9

Nemhauser, G. L., & Wolsey, L. A. (2008). *Integer and combinatorial optimization*. John Wiley & Sons. https://doi.org/10.1002/9781118627372

Nocedal, J., & Wright, S. J. (2006). *Numerical optimization*. Springer Science & Business Media.

Okposo, N.I., Addai, E., Apanapudor, J.S., and Gomez-Aguilar, J.F.(2023): A Study of Monkey-pox transmission model within the scope of fractal-fractional derivative with power-law kernel, The European Physical Journal Plus, https://doi.org/epjp/s13360-023-04334-1

Okwonu,F.Z.,Ahad, N.A.,Apanapudor,J.S.,and Arunaye, F.I.(2021):Robust Multivariate Correlation Techniques: A

Confirmation Analysis using COVID-19 Data Set, Pertanika Journal of Science and Technology, Vol. 29(2), pp.999 – 1015

Okwonu, F.Z. Ahad, N.A., Okoloko,I. E., Apanapudor, J.S.,and Arunaye, F.I. (2022): Robust Hybrid Classification Methods and Applications, Pertanika Journal of Science and Technology,Vol. 10(4), pp. 2831 - 2850.

Okwonu, F. Z. Ahad, N.A., Apanapudor, J.S. and Arunaye, F.I.(2023): Chi-square and Adjusted Standardised Residual Analysis,ASM Sc.Journal Vol. 18, https://doi.org/10.32802/asmscj.2023.985

Ravindran, A., Phillips, D. T., & Solberg, J. J. (2013). *Operations research: principles and practice*. John Wiley & Sons.

Sivanandam, S., Sumathi, S., & Deepa, T. (2008). *Introduction to Genetic Algorithms*. Springer.

Shi, Y., & Eberhart, R. C. (2008). A modified particle swarm optimizer. In Proceedings of the IEEE International Conference on Evolution

Voss, S. (2010). Evolutionary algorithms: The key to solving complex optimization problems. *Applied Soft Computing, 10*(2), 1–10.

Wang, G. G. (2013). A hybrid evolutionary algorithm for complex engineering optimization problems. *Engineering Optimization, 45*(3), 291–309.

Zhang, Y., Liu, X., & Ma, L. (2021). Blockchain in supply chain management: A literature review and future research directions. *Journal of Business Logistics, 42*(1), 14–26

Zhou, Z., Li, X., & Jiang, J. (2021). The role of blockchain technology in the fight against COVID-19: A review. *International Journal of Information Management, 44*, 74–79.

Zitzler, E., Laumanns, M., & Thiele, L. (2011). SPEA2: Improving the strength Pareto evolutionary algorithm. TIK-Report, (103).