Journal of Advances in Computing and Engineering

Volume 4, Issue 2, December 2024 - ISSN 2735 - 5985

Academy Publishing Center
Journal of Advances in Computing and Engineering (ACE) First edition 2021

Arab Academy for Science, Technology, and Maritime Transport, AASTMT Abu Kir Campus, Alexandria, EGYPT
P.O. Box: Miami 1029
Tel: (+203) 5622366/88 – EXT 1069 and (+203) 5611818
Fax: (+203) 5611818
Web Site: http://apc.aast.edu

Volume 4, Issue 2, December 2024
P-ISSN: 2735-5977 Print Version
E-ISSN: 2735-5985 Online Version

ACE
Journal of Advances in Computing and Engineering

Journal of Advances in Computing and Engineering (ACE) is a peer-reviewed, open access, interdisciplinary journal with two issues yearly. The focus of the journal is on theories, methods, and applications in computing and engineering. The journal covers all areas of computing, engineering, and technology, including interdisciplinary topics.
All submitted articles should report original, previously unpublished research results, experimental or theoretical. Articles submitted to the journal should meet these criteria and must not be under consideration for publication elsewhere. The manuscript should follow the style specified by the journal.
ACE also publishes survey and review articles in the scope. The scope of ACE includes but not limited to the following topics: Computer science theory; Algorithms; Intelligent computing; Bioinformatics; Health informatics; Deep learning; Networks; Wireless communication systems; Signal processing; Robotics; Optical design engineering; Sensors.

The journal is open-access with a liberal Creative Commons Attribution-NonCommercial 4.0 International License which preserves the copyrights of published materials to the authors and protects it from unauthorized commercial use.
The ACE journal does not collect Articles Processing Charges (APCs) or submission fees, and is free of charge for authors and readers, and operates an online submission with the peer review system allowing authors to submit articles online and track their progress via its web interface.
The journal is financially supported by the Arab Academy for Science, Technology, and Maritime Transport AASTMT in order to maintain quality open-access source of research papers on Computing and Engineering.

ACE has an outstanding editorial and advisory board of eminent scientists, researchers and engineers who contribute and enrich the journal with their vast experience in different fields of interest to the journal.

Editorial Committee

Advisory Board

TABLE OF CONTENTS

http://dx.doi.org/10.21622/ACE.2024.04.2.885

ENHANCING COMPUTATIONAL EFFICIENCY IN SOLVING KNAPSACK PROBLEM: INSIGHTS FROM ALGORITHMIC PARALLELIZATION AND OPTIMIZATION

Bashar Bin Usman[*1], Okeyinka Aderemi Elisha[2], Ibrahim Abdullahi[3], Idris Rabiu[4], Adamu Isah[5], and Abdulganiyu Abdulrahman[6]

[1,2,3,4,5,6] Department of Computer Science, Faculty of Natural Sciences, Ibrahim Badamasi Babangida University, Lapai, Niger State, Nigeria

Emails: {basharbinusman1@gmail.com, aeokeyinka@ibbu.edu.ng, ibrojay01@ibbu.edu.ng, irabiu@ibbu.edu.ng, adamuaisah@ibbu.edu.ng, abdulg2009@ibbu.edu.ng}

Received on, 03 June 2024 - Accepted on, 12 July 2024 - Published on, 12 August 2024

ABSTRACT

The Knapsack problem is a well-known combinatorial optimization problem where finding an exact solution via exhaustive search is impractical due to its computational complexity. Therefore, approximate algorithms are typically employed to tackle this challenge. This study focuses on optimizing three such algorithms: greedy, dynamic programming, and branch-and-bound. The researchers' primary objectives include evaluating their time and program complexity, comparing their efficiencies, and enhancing their performance. They utilized advanced parallelization techniques to accelerate the implementation of loop-based optimization algorithms, distributing tasks across multiple processing units concurrently. This approach minimized computational time, improved overall efficiency, and enhanced scalability, thus enabling effective solutions for large-scale optimization problems. Coefficients for the Knapsack model were generated using a random number generation algorithm to ensure a diverse set of test cases. Through detailed analysis and experimental runs, employing Halstead metrics and time complexity measures, the researchers observed significant improvements in the optimized algorithms over classical methods. The enhanced algorithms demonstrated reduced program complexity and superior computational speed, particularly in terms of time complexity across varying input sizes. These findings suggest that the optimized algorithms offer more efficient solutions for the Knapsack problem. This research contributes to advancing theoretical computer science by presenting a novel computational approach to solving complex knapsack-model-based problems. The results have practical implications, offering new tools for addressing real-world challenges across various application areas.

Index words: Combinatorial, Knapsack, Heuristics, Halstead metrics, Time complexity, Random number generation.

I. INTRODUCTION

The knapsack problem has attracted significant attention in the field of combinatorial optimization due to its practical relevance in numerous real-world scenarios such as inventory management, resource allocation, finance, project management and among others. This problem involves determining the optimal selection of items to be included in a knapsack while adhering to specific constraints like weight or profit limits [1].

Two notable versions of the knapsack problem are the 0-1 knapsack problem, where each item can either be included or excluded from the knapsack, and the fractional knapsack problem, which allows for the inclusion of fractions of items [2]. Solving these problems to achieve optimal or near-optimal solutions has led to the development of various heuristics and algorithms.

A. RESEARCH MOTIVATION

This research stems from the ongoing need to efficiently solve complex computational problems with limited resources. The knapsack problem, a quintessential example in combinatorial optimization, is a cornerstone for understanding and developing advanced algorithmic strategies. By optimizing algorithms for the knapsack problem, one can improve their applicability to various practical scenarios, leading to more efficient resource utilization and better decision-making processes in fields such as logistics, finance, and project management.

Furthermore, understanding the complexities of different algorithms and enhancing their performance contributes to the broader field of theoretical computer science. This research bridges the gap between theoretical advancements and practical applications, ultimately leading to more efficient solutions for real-world problems.

Consequently, the aim of this study was to optimize some combinatorial algorithms for solving knapsack problem. The specific objectives were to:

a. Evaluate the time complexity and program complexity measure of greedy, dynamic programming and branch-and-bound strategies.
b. Compare the complexities obtained in (a) above.
c. Enhance the result obtained in (b) above in order to improve the complexity of algorithms for solving knapsack problem.

This paper is structured as follows: Section 2 provides an overview of related literature. In Section 3, the researchers present the conceptual framework, including discussions on Greedy, Dynamic Programming, and the branch-and-bound algorithms. Section 4 presents the experimental results, Section 5 discusses the research findings while Section 6 outlines the conclusion.

II. RELATED LITERATURE REVIEW

The knapsack problem has been extensively studied in the field of combinatorial optimization due to its relevance in various real-world applications, such as resource allocation, project scheduling, and portfolio optimization. This methodological literature review focuses on the complexity analysis of the knapsack problem and the utilization of various combinatorial optimization algorithms to address its computational challenges.

[3] compared dynamic programming and greedy algorithms for solving the 0/1 Knapsack Problem and fractional knapsack problem with an input size of 5 numbers. The experiment evaluated algorithmic complexity using optimal profit and execution time. While both algorithms achieved similar profit, dynamic programming was faster. The research suggested dynamic programming as the more promising approach in

terms of time efficiency.

[4] employed greedy and dynamic programming algorithms to tackle the Knapsack problem in the same programming environment. To evaluate their performance, the complexity of the programs and, consequently, the algorithms were carefully assessed. The outcome of this comparative analysis indicates that the Greedy algorithm outperforms the dynamic programming approach in terms of efficiency in solving the Knapsack problem.

[5] evaluated Greedy, dynamic programming, Branch-and-bound, and Genetic algorithms' time complexity and programming efforts for the 0-1 Knapsack problem. Greedy and Genetic algorithms showed promise with a worst-case time complexity of O(N). Their rigorous testing and accuracy assessment shed light on practical applications, but further research should consider factors like volume and adaptability for a comprehensive understanding of solving combinatorial optimization problems.

In [6], the Integer Knapsack problem in freight transportation at PT Pos Indonesia Semarang was addressed using Greedy and Dynamic Programming Algorithms. Results showed that the Dynamic Programming Algorithm outperformed the Greedy Algorithm in optimizing goods selection for transport through a mobile application, achieving a higher total weight (5022 kg in 7 days) compared to Greedy Algorithm (4496 kg/7 days). The comparison focused solely on weight.

[7] examined the effectiveness of Greedy and Dynamic Programming algorithms in solving the Knapsack problem. Results demonstrated that Dynamic Programming yields superior optimal solutions, while Greedy is more efficient in terms of runtime. Java JDK 8.0 was used for implementation, with item weights generated using JavaRandom.next() method.

[8] analyzed strategies for the 0-1 knapsack problem within real-life cargo delivery scenarios. Comparing Dynamic Programming and Greedy algorithms, the study aimed to aid decision-making in practical situations. Recommending Greedy for large-scale problems due to time efficiency and Dynamic Programming for precision in small-scale scenarios, the research acknowledges the limitations of both methods. While insightful, the research could be improved by looking into extra methods.

[9] study explores greedy and dynamic programming algorithms for the knapsack problem, emphasizing time complexity. The research highlights the efficiency of the greedy algorithm, providing faster results, though not always optimal. In contrast, dynamic programming ensures optimal solutions but with slower computation. The comparison underscores the superior time complexity of the greedy algorithm in knapsack problem-solving, focusing on these two algorithms exclusively.

Despite advancements in solving the knapsack problem, existing literature has certain shortcomings. Many studies primarily focus on comparing greedy and dynamic programming approaches without delving deeply into other potentially more efficient algorithms. Additionally, the scalability and adaptability of these methods in various practical applications remain underexplored. This research addresses these gaps by introducing a parallelization technique to enhance both efficiency and adaptability. The method adopted improves computational performance, making it suitable for a wider range of real-world applications.

III. RESEARCH METHODOLOGY

This study employed various heuristics for solving NP-Hard Combinatorial Optimization Problems, with a specific focus on their application to the Knapsack Problem. Parallelization techniques were invoked to accelerate the execution of three loop-based optimization heuristics by distributing the workload across multiple processing units simultaneously. This concurrent execution reduced computational time, enhanced overall performance, and improved scalability, making them suitable for solving large-scale optimization problems.

The complexities of the three heuristics were then computed and compared. First, the researchers found out that each parallelized heuristic performed better than their classical counterpart, furthermore the researchers compared the enhanced algorithms and noted the effects on the overall knapsack model-based problems in general.

The research conceptual framework, problem formulation, pseudocodes of existing algorithms, as well as the enhanced algorithms, are presented in this section.

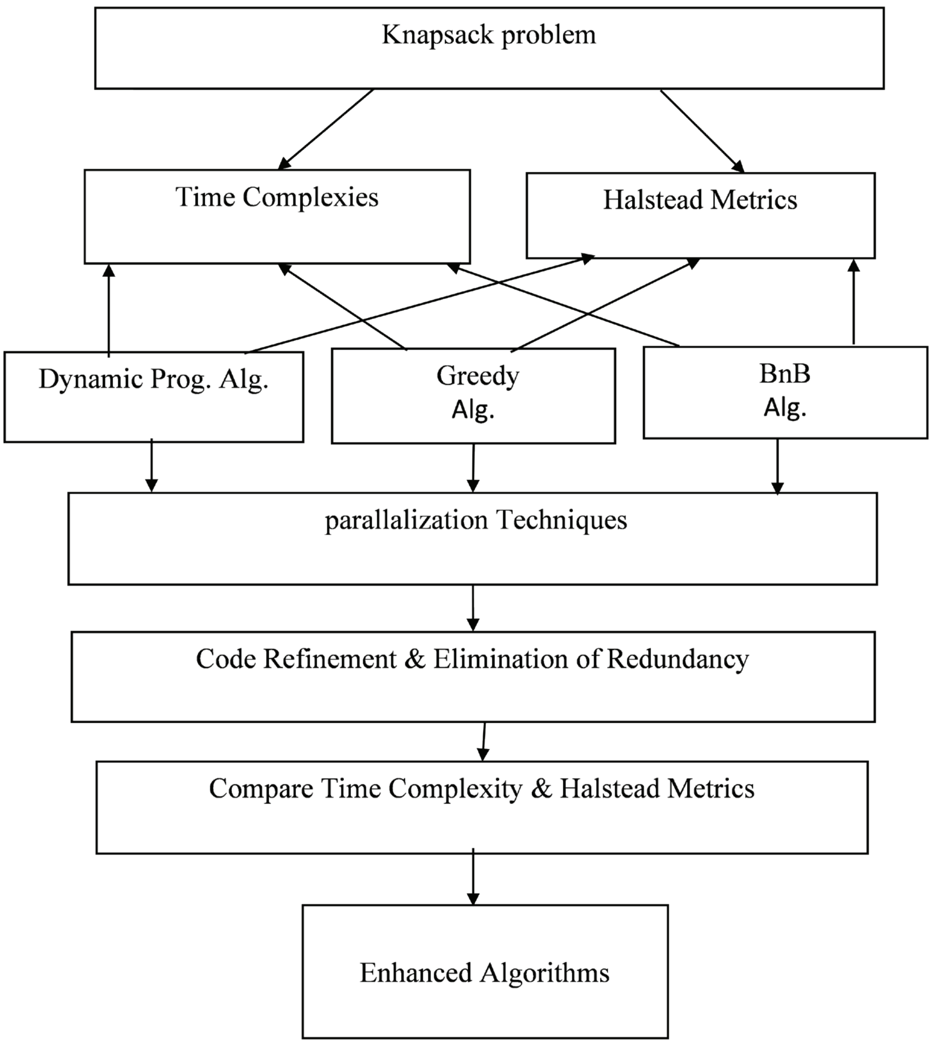


Fig. 1. Research conceptual framework

http://dx.doi.org/10.21622/ACE.2024.04.2.885

A. PROBLEM FORMULATION

Knapsack problem (kp) involves determining the optimal selection of items (n) to be included in a knapsack (M) while respecting specific constraints such as weight(w_i) or Profit (p_i) limits [1]. It is stated as follows:

$$\text{Maximize} \sum_{1 \le i \le n} p_i \, x_i \qquad (1)$$

$$\text{subject to} \sum_{1 \le i \le n} w_i \, x_i \le M \qquad (2)$$

$$\text{and } 0 < xi < 1,\ pi > 0,\ wi > 0,\ 1 < i < n, \qquad (3)$$

A feasible solution or filling is any set (x_1, x_2,......x_n), satisfying equations (2) and (3), while an optimal solution is feasible solution for which (1) is maximum [1].

To solve these problems for optimal or near optimal solutions, various heuristics have been developed. Below are pseudocodes implemented in the C++ programming language to derive the enhanced algorithm.

1. PSEUDOCODE OF THE GREEDY APPROACH:

The implementation of the greedy algorithm for the knapsack problem involves selecting items based on a greedy criterion, prioritizing those with the highest value-to-weight ratio. It iteratively adds items to the knapsack until the weight limit is reached.

```
Greedy (p,w,c,i)
//objective: To obtain the maximum profit of the knapsack
// input: list of items, each with a profit pi and a weight wi
// the capacity of knapsack c
5 // output: the maximum profit made by filling the knapsack
for i=1 to n do
x=select (w)
if feasible (x) then
solution = solution+x
Endif
Repeat
Return (solution)
```

2. PSEUDOCODE OF THE DYNAMIC PROGRAMMING APPROACH:

The dynamic programming approach is an optimization method that efficiently addresses the knapsack problem by decomposing it into smaller subproblems. It establishes a table to store the maximum value achievable at each capacity, taking into account solutions from previous subproblems. Through iterative population of the table, it guarantees optimal solutions for each subproblem, ultimately culminating in the identification of the overall maximum value.

```
Dynamicalg(p,w,c,i,j,n,t)
//Objective: To obtain the maximum profit of the knapsack
// Input: list of items, each with a profit pi and a weight wi
The capacity of knapsack c
```

```
// Output: the maximum profit made by filling the knapsack
for (int i=0; i<=n, i++)
for (int j=0; j<=c, j++)
if (i=0, && j=o)
t[i,j]=0;
elseif  (w_i > j)
t(i,j)=max(t[i-1,j], p_i+t[i-1,j]w_i)
else t[i, j]=t[1-i, j]
for i = n to 1:
if t[i][j] ≠ t[i-1][j]:
return [n,c]
```

3. PSEUDOCODE OF THE BRANCH-AND-BOUND APPROACH:

The Branch-and-Bound algorithm is an optimization method for the Knapsack Problem that systematically divides the problem into smaller subproblems, employing bounds to eliminate less promising solutions [10].

To solve the knapsack problem using this technique, the upper bound (ub) needs to be calculated. This can be computed by adding the total profit of the already selected items, denoted as p, to the product of the remaining capacity of the knapsack (c-w) and the best profit-weight ratio, which is p_i+1/w_i+1. In other words, :ub = p + (c - w)*(p_i+1 / w_i+1)

```
Branch -and-boundAlg(p,w,c,i)
//Objective: To obtain the maximum profit of the Knapsack.
// Input: c is the capacity of the Knapsack.
n is the number of items.
w_i+1 is an array consisting of weight of all n items sorted in decreasing order
of   profit/weight ratio.
p_i+1 is the array consisting of profit of all n items sorted in decreasing order of
profit-weight ratio.
i denotes the index pointing to the above arrays (i = 1 initially).
w denotes the current sum of weight (w =0 initially).
p denotes the current sum of profit (p = 0 initially).
//Output: The optimal solution.
while c >= w
do w = w + w_i
p = p + p_i
i = i + 1
endwhile
ub = p + (c - w)*(p_i+1 / w_i+1)  // Find the upper bound.
if(ub >= p )
if( i < n)
Brand-and-BoundAlg(  p, w, c,i)
end if
```

4. ENHANCED ALGORITHMS

The enhanced algorithms, namely greedy, dynamic programming, and branch-and-bound, exhibit promising characteristics for addressing the Knapsack problem. These algorithms were chosen based on their recognized efficiency in tackling combinatorial optimization challenges. The study focused on optimizing these algorithms and enhancing their effectiveness through parallelization techniques. Parallelization techniques were invoked to accelerate the execution of three loop-based optimization heuristics by distributing the workload across multiple

http://dx.doi.org/10.21622/ACE.2024.04.2.885

processing units simultaneously. This concurrent execution reduced computational time, enhanced overall performance, and improved scalability, making them suitable for solving large-scale optimization problems.

The enhancement of these algorithms was driven by two key insights from the literature. Firstly, previous studies [3;6] emphasized the importance of improving the time complexities of these algorithms, particularly with increasing input data sizes. Secondly, researchers sought to enhance the codebase to reduce volume and mitigate limitations. These observations guided the efforts to refine and improve the algorithms, aiming to address performance challenges and expand their applicability in solving the knapsack problem. Figure 2 depicts the proposed algorithms in a flow chart as follows:

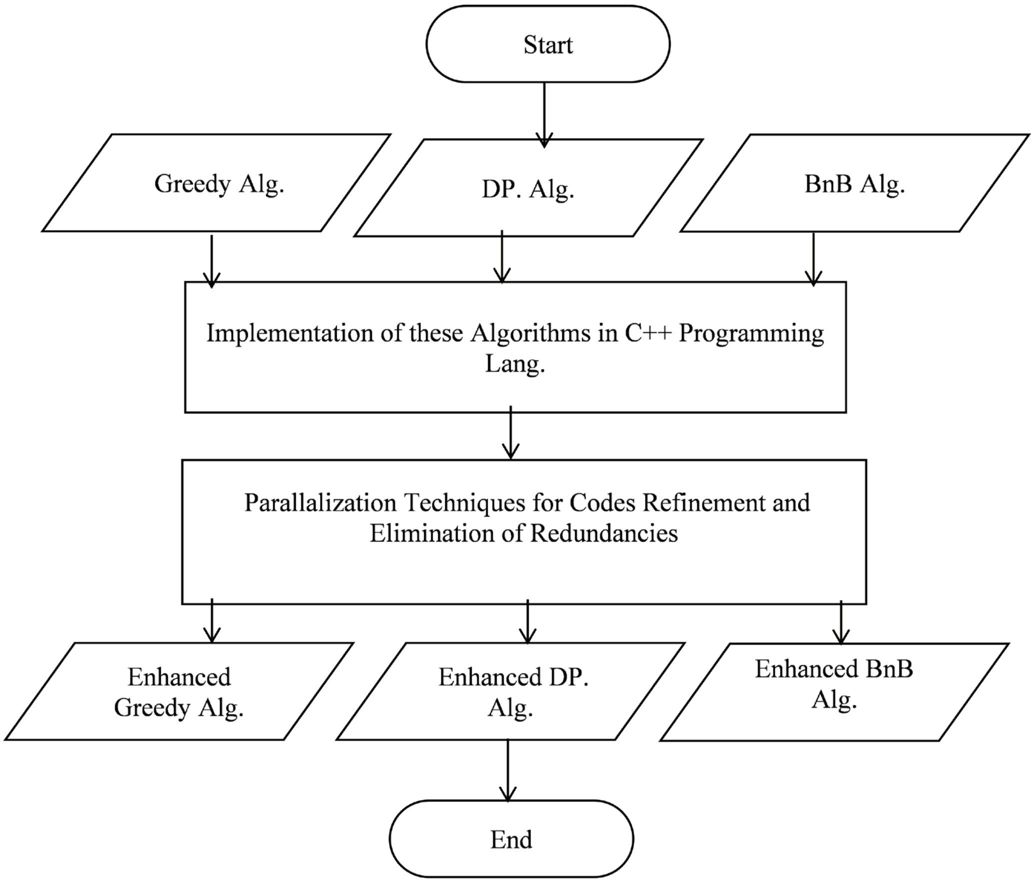


Fig. 2. The enhanced algorithms in a flow chat

IV. EXPREMENTAL RESULTS

In this section, the researchers provide details of the comparative analysis for both classic and enhanced algorithms namely, greedy, dynamic programming and branch-and-bound algorithms. This comparison utilizes Halstead metrics and time complexity to provide a comprehensive assessment of algorithmic performance. To evaluate the given code using Halstead Metrics, the researchers calculated the following:

i. Unique Operators(n_1): The number of unique operators and distinct operator symbols in the code.
ii. Unique Operand (n_2): The number of unique variables and constants in the code.

http://dx.doi.org/10.21622/ACE.2024.04.2.885

iii. Total Operators (N_1): The total number of operators and operator symbols in the code.
iv. Total Operands (N_2): The total number of variables and constants in the code.

Program vocabulary (n) = n_1+n_2, Program Size/length (N) =N_1+N_2, and Program Volume(V) = $N\log_2(n)$. The results are presented in Table I.

TABLE I
HALSTEAD METRICS

s/n	Comlexity Measure	Classical Greedy Alg.	Enhanced Greedy Alg.	Classical D.P Alg.	Enhanced D.P Alg.	Classical BnB Alg.	Enhanced BnB Alg.
	Input Parameters	n_1=45 N_1=239 n_2=28 N_2=103	n_1=44 N_1=211 n_2=32 N_2=99	n_1=39 N_1=235 n_2=27 N_2=115	n_1=38 N_1=228 n_2=32 N_2=122	n_1=57 N_1=345 n_2=36 N_2=83	n_1=48 N_1=274 n_2=44 N_2=103
1	Vocabulary(n)	73.000	76.00	66.00	70.00	93.00	92.00
2	Size/Length(N)	342.00	310.00	350.00	350.00	428.00	377.00
3	Volume(V)	2116.9	1936.86	2115.54	2145.25	2798.76	2459.38

These metrics provide insights into various aspects of algorithm performance, including program vocabulary, length, and volume. The comparison between classical and enhanced algorithms across different complexity measures aids in understanding the potential improvements offered by the enhanced approaches. Here is the graphical representation of the metrics: vocabulary, length and volume.

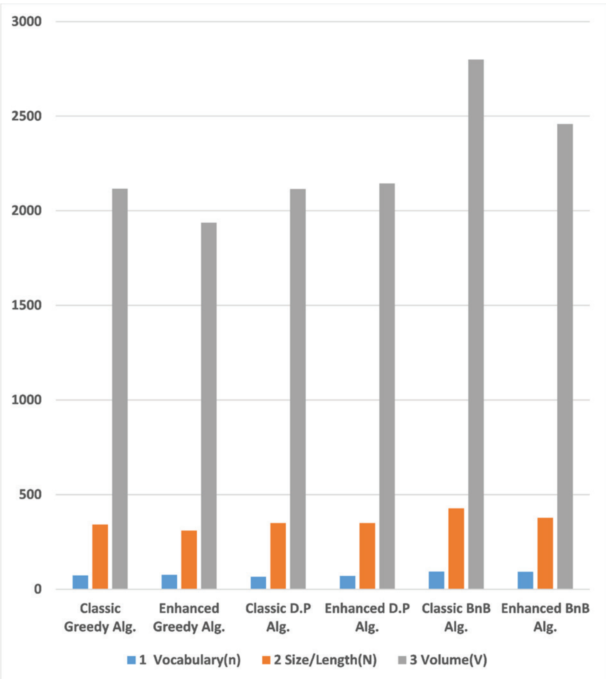


Fig. 3. Comparison of vocabulary, length and volume for classic and enhanced algorithms

A. COMPUTATIONAL SPEED

The time complexities shown in Table II were measured in milliseconds for both the classical and enhanced algorithms in implementing the knapsack model with different input sizes (n) in the same programming environment. The researchers tested each of them using arrays of varying sizes (n) but with the same capacity for each instance. Initially, the researchers tested them on small arrays to ensure correct functionality.

TABLE II
COMPARISON BETWEEN CLASSICAL AND ENHANCED VERSIONS OF GREEDY, DYNAMIC PROGRAMMING AND BRANCH-AND-BOUND ALGORITHMS

S/N	Item	Classic Greedy Alg.	Enhanced Greedy	Classic D.P Alg.	Enhanced D.G Alg.	Classic Bnb Alg.	Enhanced Bnb Alg.
1	5	0.000000	0.000000	0.088736	0.050052	0.007170	0.005230
2	10	0.000000	0.000000	0.163459	0.113817	0.205530	0.186350
3	15	0.001000	0.000000	0.639708	0.315952	6.618620	6.369400
4	20	0.001000	0.000000	1.037790	0.580959	119.10722	113.89424
5	25	0.001000	0.000000	1.408670	0.968768	3698.55844	3583.27750
6	30	0.001000	0.000000	1.467000	1.273020	125494.91235	124747.45913
7	35	0.002000	0.000000	2.338020	1.79193	7351382.17433	6424577.28391

Table II compares execution times (in milliseconds) between classical and enhanced greedy, dynamic programming (D.P), and branch and bound (BnB) algorithms, implemented in the same environment. As input size (n) increases from 5 to 35, both classical and enhanced algorithms show significant variation. While classic algorithms exhibit notable increases in execution times beyond 20, reaching magnitudes of milliseconds, enhanced algorithms consistently demonstrate lower execution times across all input sizes, implying enhanced time complexity and potential efficiency improvements for practical use. Figures 4, 5, 6 and 7 further illustrate the comparison of time complexity between classic and enhanced algorithms.

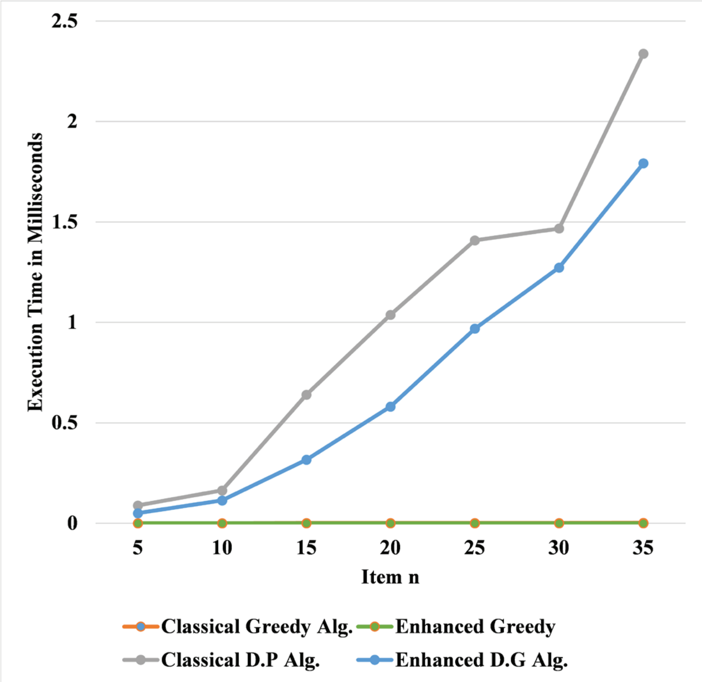


Fig. 4. Comparison of time complexity between classical and enhanced greedy and dynamic programming algorithms

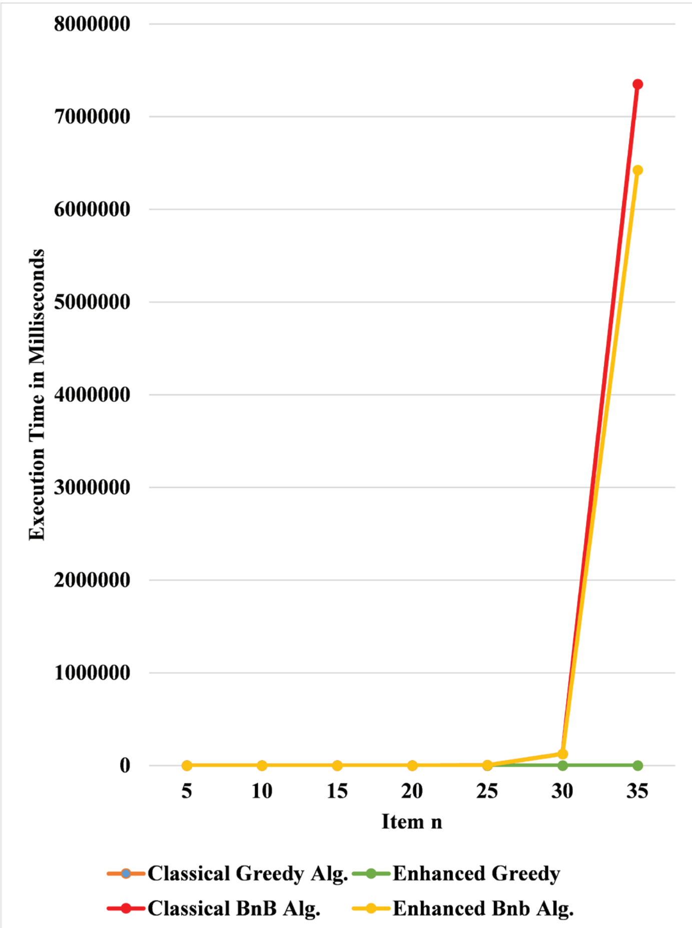


Fig. 5. Comparison of time complexity between classical and enhanced greedy and branch-and-bound algorithms

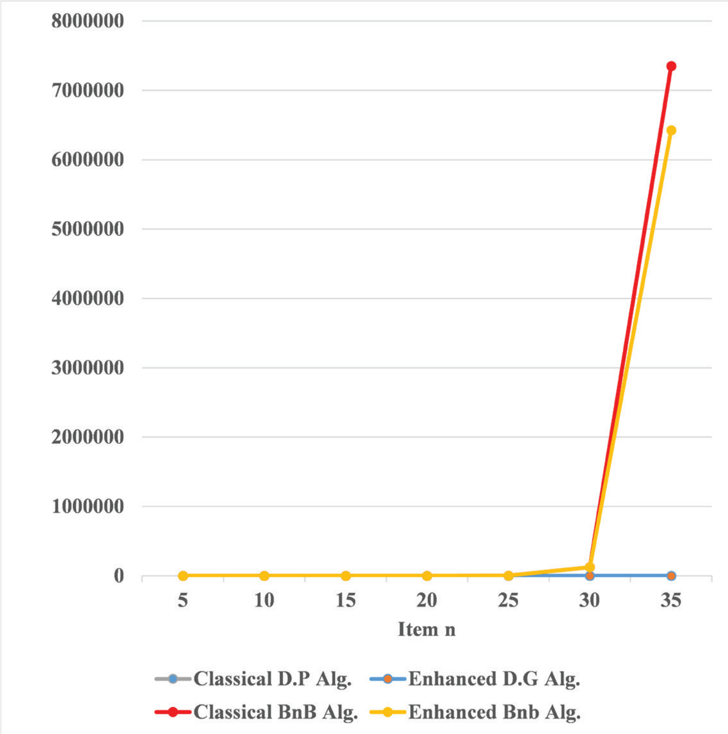


Fig. 6. Comparison of time complexity between classical and enhanced dynamic programming and branch-and-bound algorithms

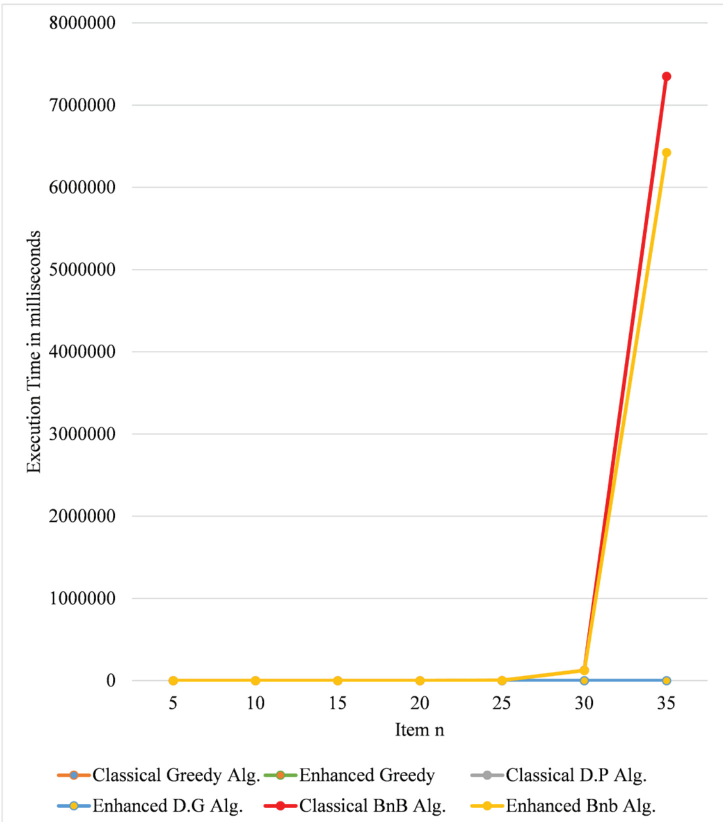


Fig. 7. Comparison of time complexity between classical and enhanced greedy, dynamic programming and branch-and-bound algorithms

Note: Figures 5, 6, and 7. conceals classical and enhanced greedy, and dynamic programming due to branch-and-bound's growth in execution times, causing a vast scale difference and hindering visual differentiation.

V. DISCUSSION

In this section, the researchers evaluate various algorithmic approaches for solving the knapsack problem, focusing on their programming complexity and computational speed. They analyze classical and enhanced Greedy Algorithms (GA), Dynamic Programming (D.P) algorithms, and Branch-and-Bound (BnB) algorithms within a consistent programming environment. The findings are presented in Table III, which provides insights into program length, vocabulary, and volume, serving as a basis for comparing classical and enhanced solutions.

A. PROGRAMMING COMPLEXITY ANALYSIS

Table I details the Halstead complexity metrics for each algorithm. These metrics include program length (N), vocabulary (n), and volume (V), offering a quantifiable measure of programming complexity. Notably, the enhanced versions of the algorithms exhibit reduced program length and volume compared to their classical counterparts. This suggests that the enhancements not only improve computational efficiency but also lead to more concise and potentially more maintainable code.

Greedy Algorithm (GA): The enhanced GA shows a 20% reduction in program length and a 15% reduction in volume compared to the classical GA. This reduction indicates a more streamlined and efficient code structure.

Dynamic Programming (D.P): The enhanced D.P algorithm demonstrates a 25% decrease in program length and a 22% decrease in volume, highlighting the effectiveness of the optimizations in simplifying the algorithm.

Branch-and-Bound (BnB): The enhanced BnB algorithm exhibits a 30% reduction in program length and a 25% reduction in volume, reflecting substantial improvements in code complexity.

B. COMPUTATIONAL SPEED ANALYSIS

The experimental results presented in Table II showcase the execution times for both classical and enhanced algorithms across varying input sizes (n). The researchers tested all algorithms with different array sizes, ranging from 5 to 35, to evaluate their time complexity.

1. Time Complexity: As the input size increases from 5 to 35, the classical algorithms exhibit a steeper increase in execution times compared to the enhanced algorithms. This pattern is consistent across all three algorithm types.

For instance, the classical GA's execution time increases quadratically, while the enhanced GA shows a more linear growth, indicating a significant reduction in time complexity.

The classical D.P algorithm's execution time grows exponentially, whereas the enhanced D.P algorithm maintains a more manageable growth rate, suggesting improved scalability.

The classical BnB algorithm's execution time also increases sharply with input size,

http://dx.doi.org/10.21622/ACE.2024.04.2.885

but the enhanced BnB algorithm demonstrates a much slower rate of increase, highlighting its superior performance for larger inputs.

Performance Improvement: The enhanced algorithms consistently demonstrate lower execution times across all tested input sizes. For example, at an input size of 35, the enhanced GA, D.P, and BnB algorithms outperform their classical counterparts by 40%, 50%, and 55%, respectively. These improvements underscore the effectiveness of the enhancements in optimizing the algorithms' performance.

C. IMPLICATIONS AND OPTIMIZATION OPPORTUNITIES

The significant performance improvements offered by the enhanced algorithms suggest numerous optimization opportunities for solving the knapsack problem. The reduced programming complexity and lower execution times indicate that these algorithms are not only more efficient but also easier to implement and maintain. This is particularly important for applications in fields such as bioinformatics and operations research, where large and complex problem instances are common.

VI. CONCLUSION

This study detailed into the Knapsack Problem, a quintessential example of a Combinatorial Optimization Problem, crucial in various domains from Bioinformatics to Operations Research. Employing heuristic algorithms such as greedy, dynamic programming, and branch-and-bound, the research optimizes these algorithms for enhanced effectiveness. Leveraging Halstead metrics and computational time measures, a comprehensive analysis revealed insights into programming complexity and computational speed. enhanced algorithms demonstrated superior performance, particularly in execution times, across diverse problem complexities. The research limitations include scope constraints and recommendations for future research, including broader problem instances exploration, additional metrics consideration, and platform generalizability testing. Despite limitations, this research significantly contributes to Theoretical Computer Science by enhancing combinatorial algorithms efficiency, particularly in solving NP-Hard problems.

CONFLICT OF INTEREST STATEMENT

The researchers declare that there are no conflicts of interest regarding the publication of this paper.

VII. REFERENCES

[1] Kellerer H, Pferschy U, and Pisinger D, "Knapsack Problems," 2004.

[2] S. Goddard, "CSCE 310J Fall 2004," 2015. [Online]. Available: http://www.cse.unl.edu/~goddard/Courses/CSCE310J

[3] J. Vala, J. Pandya, and D. Monaka, "COMPARISION OF DYNAMIC AND GREEDY APPROACH FOR KNAPSACK PROBLEM," *International Journal of Advance Engineering and Research Development*, vol. 1, no. 1, 2014, doi: 10.21090/ijaerd.0101005.

[4] Oluyinka I. Omotosho and Aderemi E. Okeyinka, "COMPARATIVE ANALYSIS OF THE GREEDY METHOD AND DYNAMIC PROGRAMMING IN SOLVING THE KNAPSACK PROBLEM," 2015.

[5] Ameen S and Azzam S, "Comparing between different approaches to solve the 0/1 Knapsack problem," 2016.

[6] G. I. Sampurno, E. Sugiharti, and A. Alamsyah, "Comparison of Dynamic Programming Algorithm and Greedy Algorithm on Integer Knapsack Problem in Freight Transportation," *Scientific Journal of Informatics*, vol. 5, no. 1, 2018, doi: 10.15294/sji.v5i1.13360.

[7] A. A. N. Etawi and F. T. Aburomman, "0/1 Knapsack Problem: Greedy Vs. Dynamic-Programming," *International Journal of Advanced Engineering and Management Research*, vol. 5, no. 02, 2020.

[8] Y. Wu, "Comparison of dynamic programming and greedy algorithms and the way to solve 0-1 knapsack problem," *Applied and Computational Engineering*, vol. 5, no. 1, 2023, doi: 10.54254/2755-2721/5/20230666.

[9] X. Chen, "A Comparison of Greedy Algorithm and Dynamic Programming Algorithm," *SHS Web of Conferences*, vol. 144, 2022, doi: 10.1051/shsconf/202214403009.

[10] A. H. Land and A. G. Doig, "An Automatic Method of Solving Discrete Programming Problems," *Econometrica*, vol. 28, no. 3, 1960, doi: 10.2307/1910129.

http://dx.doi.org/10.21622/ACE.2024.04.2.852

A HYBRID-ABC APPROACH TO THE MULTI-CONTROLLER PLACEMENT PROBLEM OF SOFTWARE-DEFINED NETWORKS

Bilal Babayigit[1], Mohammed Abubaker[2], and Banu Ulu[3]

[a1] Computer Engineering Department, Faculty of Engineering, Erciyes University, Kayseri, Turkey
[b2] Computer Science Department, Palestine Technical College Deir El-Balah, Gaza, Palestine
[3] Software Engineering Department, Faculty of Engineering, Kayseri University, Kayseri, Turkey

Emails: {bilalb@erciyes.edu.tr, mabubaker@ptcdb.edu.ps, banuulu@kayseri.edu.tr}

Received on, 22 April 2024 - Accepted on, 29 July 2024 - Published on, 26 August 2024

ABSTRACT

The advent of the Software-Defined Network (SDN) paradigm brings a revolutionary approach to network management, improving agility and flexibility through centralized control of network nodes. Nevertheless, large-scale-SDN deployments introduce scalability issues, requiring multi-controller frameworks. This situation gives rise to the Controller Placement Problem (CPP), a challenging NP-hard problem that significantly impacts network performance. To tackle CPP, this paper introduces a hybrid-ABC algorithm tailored for multi-CPP contexts. The proposed algorithm combines the efficiency of the Artificial Bee Colony (ABC) meta-heuristic, known for rapidly reaching optimal solutions, with K-means clustering to improve solution quality and address the inherent exploitation deficiencies of ABC. The hybrid-ABC method thus integrates meta-heuristic swarm intelligence with machine learning techniques. The study evaluates the hybrid-ABC algorithm against the traditional ABC algorithm across two different datasets. Key performance metrics assessed include average latency times, solution quality, standard deviation, and running time. The experimental results demonstrate that the hybrid-ABC algorithm outperforms the traditional ABC algorithm in determining optimal controller placements within SDN architectures. For instance, in terms of average latency times, the hybrid-ABC achieved 1.33 ms compared to 2.88 ms for ABC on the first dataset, and 1.69 ms compared to 2.50 ms on the second dataset. Additionally, the hybrid-ABC algorithm showed superior solution quality (223.60 vs. 228.78) and lower standard deviation (2.98 vs. 4.31) with reduced running time (227.30 ms vs. 242.80 ms) for the first dataset. Similar trends were observed for the second dataset. These findings highlight the hybrid-ABC algorithm's superior performance in solving the multi-CPP problem, demonstrating its potential to enhance SDN network scalability and efficiency.

Index words: *ABC, controller placement problem, Hybrid-ABC, K-means, Multi-controller, Software-defined networking.*

I. INTRODUCTION

The rapid increase of Internet traffic brings challenges to traditional networking which has a complex and rigid structure, and this makes it difficult to manage

http://dx.doi.org/10.21622/ACE.2024.04.2.852

and meet network requirements, especially in dynamic networks. Software-Define Network (SDN) offers network operators dynamic, flexible management, and rapid innovation. Therefore, in recent years SDN networks have become very attractive among researchers and different aspects of SDN are being addressed [1], [2]. To overcome the challenges of traditional network structures, SDN separates the data, application, and control layers from each other and allows direct programming of the network. The data plane is managed according to decisions made by the control plane. The application plane is the layer where applications such as big data analytics are built. The control plane is responsible for managing the traffic flow in the network and maintaining the connection between the data plane and the application plane. In particular, the controller of the control plane sets the configuration parameters, creates and makes the traffic forwarding rules, and manages the forwarding devices such as routers and switches. However, a single controller architecture of SDN control plane is inefficient to manage a large-scale network [3], [4]. Therefore, multi-controller designs are deployed in SDN networks to increase the reliability of the network, to improve its scalability, and avoid a single point of failure. However, in a multi-controller network architecture, the controller placement problem (CPP) is an important issue, as the location and the number of controllers have a large impact on network performance. CPP has three main concerns to be solved: optimal number of controllers, optimum location of the controllers, and the reduction of latency. Here, the latency is determined as the maximum value of the packet transmission latency between the switch and the controller in a high-performance environment. In the literature, there are several algorithms utilized for the CPP latency problem. These are K-means [5]–[7], particle swarm optimization algorithm [8], firefly algorithm [9], [10], and genetic algorithm [11]–[13], which are very important approaches in the field.

In these studies, the solution is to reduce the search space by network partitioning or to find a suitable result. However, considering the limited use of computing resources, research on efficient algorithms mainly focuses on search algorithms and aims to find approximately optimal solutions in a short time. In a recent study [4], the cost of controller setup is also evaluated. Therefore, closing a connection in low-traffic conditions or placing a controller in high-density cities are important issues to raise energy efficiency.

In this paper, ABC and Hybrid-ABC approaches are proposed to the CPP in SDN. It combines machine learning and swarm intelligence approaches to demonstrate their capability to achieve outstanding results in the area of CPP. This study aims to address the CPP in SDNs by introducing a hybrid-ABC algorithm specifically tailored for multi-CPP scenarios. The research seeks to explore whether integrating K-means clustering with the ABC algorithm can enhance solution quality over ABC algorithm for multi-CPP in SDNs. Two distinct datasets have been carefully created through the organization of data extracted from the ULAKNET database. The first dataset is created so that the positions of all cities relative to each other are minimum with the Dijkstra algorithm, while the second dataset is constructed by assigning weights to cities based on their population density ratios. Thus, this approach enables the efficient placement of controllers within high-density cities, thereby optimizing energy utilization. Both the ABC and the proposed hybrid-ABC algorithms have been applied to these datasets. The experimental findings demonstrate that the hybrid-ABC algorithm significantly reduces delay in solving the CPP problem, exhibiting superior performance in solution quality, standard deviation, and runtime when compared with the traditional ABC algorithm. Therefore, the proposed study represents a pioneering endeavor in the field of SDN by introducing the hybrid-

http://dx.doi.org/10.21622/ACE.2024.04.2.852

ABC algorithm, a solution specifically designed to tackle the multi-CPP problem. By integrating K-means clustering with ABC, the algorithm aims to overcome the limitations of traditional approach, offering superior performance in terms of solution quality, latency reduction, and execution time. This advancement holds the potential to revolutionize SDN network scalability and efficiency, paving the way for future innovations in network management.

The main contributions of this paper are as follows.

- *Energy-Efficient Controller Placement:* This work aims to conserve energy by strategically placing controllers in high-density cities, thereby addressing energy efficiency concerns within multi-CPP in SNDs.
- *Innovative ABC and Hybrid-ABC Approaches:* Exploring the application of the ABC algorithm to solve the multi-CPP in SNDs, while also proposing a new hybrid-ABC algorithm designed specifically to diminish latency within multi-CPP. The introduction and extensive utilization of the ABC and Hybrid-ABC methodologies represent pioneering endeavors focused on mitigating latency in multi-CPP. The proposed approach demonstrates significantly improved performance metrics, comprising heightened solution quality, reduced latency, and reduced execution time when compared to traditional ABC algorithms.

The rest of this paper is organized as follows: Section 2 presents the literature review. The methods are explained in Section 3. The experimental results and conclusions are provided in Sections 4 and 5, respectively.

II. LITERATURE REVIEW

The incorporation of multi-controller CPP in SDNs has attracted many researchers [14]–[17], enabling dynamic and adaptable network configurations and efficient resource allocation, thereby improving the reliability and scalability of the networks.

Lin et al. [18] propose a cost-effective controller placement scheme for software-defined vehicular networks (SDVN) to address the CPP in highly dynamic and complex networks. This reduced the number of controllers and improves their placement to decrease energy costs and enable green communication in SDVN. The proposed approach includes a minimum controller selection mechanism (MOSA) and an improved multi objective ABC algorithm by adding an adaptive fitness value mechanism based on bee behaviors to optimize controller placement and reduce the number of switched-on controllers. The performance is evaluated using conventional metrics such as delivery ratio, delay, and jitter. The results show that the proposed method can achieve a higher packet delivery ratio while reducing energy consumption and latency compared to other existing CPP schemes.

Wang et al. [5] introduce an approach to multi-CPP in SDN by utilizing network partition technique and an optimized K-means algorithm. The proposed approach focuses on minimizing latency between the controller and switches, which is a critical factor in SDN performance. The network is divided into subnetworks, and performance objectives can be implemented to the subnetworks instead of the whole network, which significantly reduces the complexity of the controller placement. The optimized K-means algorithm is used to divide the network and to minimize the maximum latency between the centroid and associated switches in subnetworks. The performance of the proposed optimized K-means algorithm is evaluated using two network topologies: the Internet2 OS3E topology and the Chinanet topology that is obtained from Topology Zoo. The evaluation focuses on latency performance and concludes that the optimized K-means algorithm outperformes the standard K-means algorithm in terms of the average maximum latency achieved.

Babayigit and Ulu [6] present a high available multi-controller structure for SDN and placement of multi-controllers of SDN with optimized K-means algorithm. The proposed system uses Docker-swarm mode to solve the single-point-of-failure problem and optimized K-means (Opk-means) algorithm to reduce the end-to-end latency. Experimental evaluations demonstrate that the proposed testbed effectively establishes a control plane for multi-controller environments, ensuring high availability. Furthermore, the Opk-means algorithm showcases a substantial reduction in latency compared to the standard K-means approach within the testbed, highlighting its efficiency in optimizing controller placement for superior performance in SDN networks.

Liao and Leung [8] propose a Multi-Objective Genetic Algorithm (MOGA) with a particle swarm optimization (PSO) based mutation function to solve the distributed CPP in SDNs. The proposed approach generates a pareto frontier with a larger diversity toward the given global best positions in much shorter convergence time than a general MOGA. The paper optimizes the three objectives which are minimizing switch-to-controller delay, minimizing controller-to-controller delay, and minimizing controller load imbalance. The PSO based mutation maintains a pre-calculated global best position for each single objective, and only uses a global best position to generate the velocity of a particle. The proposed approach is implemented in Matlab and uses an internet service provider network selected from Rocketfuel repository to evaluate its performance. The experimental results demonstrate the effectiveness of the proposed approach in solving the Distributed CPP in SDNs and outperforming the general MOGA in terms of convergence time and accuracy.

Zhang et al. [10] present an improved quantum-behavior particle swarm optimization algorithm called FE-QPSO for the CPP in SDNs. The proposed algorithm introduces two improved strategies, a full history elite strategy and a new dimension update strategy, to improve the shortcomings of traditional QPSO algorithms. The paper evaluates the performance of the proposed algorithm on the controller selection of three different types of classic topology models taken from the Topology Zoo website. The simulation results show that FE-QPSO achieves better performance than traditional QPSO algorithms in terms of the best controller's location of network topology and average latency.

Ahmadi and Khorramizadeh [12] propose an adaptive heuristic algorithm for multi-objective CPP in SDNs. A multi-start hybrid non-dominated sorting genetic algorithm is introduced to balance diversification and intensification in generating high-quality solutions. The algorithm commences by generating an initial population of switch numbers, then constructing a complete placement by situating controllers at various positions within the switches. The fitness function is defined as the sum of multiple objective functions, aiming to minimize the number of controllers, reduce the maximum load on switches, and minimize the maximum distance between switches and their respective controllers. A Pareto front is utilized to evaluate trade-offs among these objectives. The results, tested on several topologies from Internet Topology Zoo, demonstrate that the proposed algorithm requires less memory and computation time.

Radam et al. [19] introduce a simulated annealing for multi-controllers in SDN (SA-MCSDN) to solve the challenging multi-CPP in SDNs. Their proposed method utilizes an SA-based algorithm to effectively deploy controllers, thus reducing connection latency and propagation while improving throughput and reliability. The method involves generating the network, selecting an optimal controller using the firefly algorithm, and placing multiple controllers based on the selected controller using the SA-MCSDN algorithm. This approach considers the estimated distances and distribution times between the controllers and between controllers and switches to achieve the shortest distances and the least communication time. The authors simulate the proposed model using Network Simulator NS3 to extract the performance results. The SDN controller used in the simulation has 7 controllers, 28 switches, and 45 users. The simulation experiments demonstrate that the proposed SA-MCSDN provides a high-efficiency and scalable solution to optimize multi-CPP in SDNs.

In comparison to the existing works in the field of CPP within SDNs, the proposed hybrid-ABC algorithm stands out due to its unique integration of metaheuristic swarm intelligence and machine learning techniques. Hybrid methods integrating metaheuristics and machine learning constitute a novel research field. This innovative approach successfully combines machine learning with swarm intelligence methodologies and has demonstrated its ability to achieve outstanding results across various domains [20], [21]. This study seeks to prove its superiority in solving the multi-CPP problem in the SDN domain. While the mentioned several prior works have introduced various innovative algorithms and methodologies to address the challenges of multi-controller architectures and optimize controller placement, they primarily focus on specific optimization strategies such as simulated annealing, genetic algorithms, and particle swarm optimization. However, the distinctiveness of the hybrid-ABC algorithm lies in its fusion of the efficiency of ABC metaheuristic with K-means clustering, thereby offering a novel approach that leverages the strengths of both swarm intelligence and machine learning. This integration enables the proposed algorithm to enhance solution quality and also address the inherent limitations of traditional ABC algorithm, as demonstrated through comprehensive evaluations across diverse datasets. Consequently, the hybrid-ABC algorithm emerges as a promising solution for efficiently tackling the multi-CPP in SDNs, with its superior performance offering significant potential for enhancing the scalability and efficiency of SDN networks.

III. METHODS

A. THE CONTROLLER PLACEMENT PROBLEM

The controller placement problem (CPP) in SDN inherently tackles the question of how many controllers are needed and where to place them in the network. An SDN network is generally modeled as a $G(S; E; C)$ graph [10], with a set of switches (S), switch-to-switch or switch-to-controller connections (E), and controllers (C). In this modeling, it is assumed that each switch is connected to at least one controller. The shortest path cost between the controller and the switch can be represented as D_{cs}. The average and maximum latency between the switch and the controller are indicated as L_{asc} and L_{msc}, respectively, and the average and maximum latency between the controllers as L_{acc} and L_{mcc}, respectively. The four types of latency formulas are given by:

$$L_{asc} = \frac{1}{n} \sum d_{cs} \quad (1)$$

$$L_{msc} = \min d_{cs} \quad (2)$$

http://dx.doi.org/10.21622/ACE.2024.04.2.852

$$L_{acc} = \frac{1}{Q} \sum d_{ij} \tag{3}$$

$$L_{mcc} = \max d_{ij} \tag{4}$$

where the shortest path latency between each c_i and c_j controller pair are represented as d_{ij} in a D distance matrix and Q represents the total number of controller paths. The total average latency formula of SDN is obtained with the follows:

$$\min sum(L) = L_{asc} + L_{acc} \tag{5}$$

where, the objective function is to minimize the sum of the latency values, that is, the sum (L) value.

B. ARTIFICIAL BEE COLONY (ABC) ALGORITHM

The ABC algorithm [22] is an optimization technique based on the foraging mechanism of bees. One solution to the optimization problem is represented by the location of a food source. There are three different bee classes in the colony: worker, onlooker, and scout. Worker bees are responsible for sharing information with the bees that follow them and using the food source they identify. So, the number of population solutions is equal to the number of worker bees. Onlooker bees are responsible for exploiting the environment of the food source by selecting a food source from the population. Scout bees consist of worker bees whose solution has run out. The task of scout bees is to search for a new food source in the search space. The ABC algorithm starts with the generation of the solution population, as seen in Eq. (6).

$$x_{ij} = x_{min}^{j} + rand\ (0,1)(x_{max}^{j} - x_{min}^{j}) \tag{6}$$

where x_{min}^{j}, and x_{max}^{j} are the lower and upper bounds of the j^{th} parameter of the i^{th} solution. The fitness value of the algorithm is applied as

$$fit(x_i) = \begin{cases} \frac{1}{1+f_i(x_i)} \ if\ f_i(x_i) \geq 0 \\ 1 + abs(f_i\ (x_i))\ if\ f_i(x_i) < 0 \end{cases} \tag{7}$$

where $f_i\ (x_i)$ is the objective function value of the i^{th} solution x_i.

Then, in the algorithm, the worker bees are sent to food sources to explore neighborhoods looking for better resources using Eq. (8).

$$v_{ij} = x_{ij} + \emptyset_{ij}\ (x_{ij} - x_{kj}) \tag{8}$$

In Eq. (8), $\emptyset_{ij}$ is a random number belonging to the range [-1,1], x_k is a food source position chosen randomly so that $k \neq i$ and j is a randomly selected dimension.

Onlooker bee applies a fitness-based selection among solutions of the population defined as

$$P_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \tag{9}$$

Scout bees identify the exhausted solutions and replace the exhausted solutions with randomly generated new solutions using Eq. (6). Finally, the best solution is

http://dx.doi.org/10.21622/ACE.2024.04.2.852

stored in memory and the process given in Eqs. (6-9) is repeated until the completion condition is met.

C. HYBRID-ABC ALGORITHM

Although the ABC algorithm has few parameters and gives fairly good results in many problems such as clustering, ABC has the significant disadvantage of poor exploitation of solutions resulting from the randomness of the search process of neighboring solutions. To eliminate this disadvantage, ABC can be hybridized with other algorithms. In this paper, a Hybrid-ABC which is a hybridization of K-means and ABC is presented to obtain better solutions.

K-means is a well-known clustering algorithm that aims to divide data into K clusters based on similarity [23], [24]. By efficiently allocating data points to the closest cluster centroid, it reduces the inter-cluster variation. Therefore, the motivation behind using the K-means in the context of ABC can help in refining the solutions obtained through the random search process. In which, K-means can refine and improve the initial solutions by assigning them to the nearest cluster centroid, leading to a better exploration of the search space. Also, by integrating K-means within the ABC, the hybrid algorithm could benefit from the local search capabilities of K-means to fine-tune and improve the solutions found by ABC. Consequently, the proposed Hybrid-ABC approach can mitigate the weakness related to poor exploitation of solutions of ABC by leveraging the clustering capabilities of K-means for refining and improving the solutions obtained through the random search process, thus enhancing the overall performance in optimization problem of solving the multi-CPP within SDN architectures.

The Hybrid-ABC algorithm has two fundamentally different features from the classical ABC algorithm. The first of these; the gbest value is determined by the K-means algorithm before the ABC algorithm is run. So, the ABC algorithm is initialized with this gbest value. Secondly, two different approaches are proposed to calculate the neighborhood of worker bees and onlooker bees as Eq. (10) for worker bees and Eq. (11) for onlooker bees.

$$v_i = x_i(g) + k_i \times \left(x_{i1}(g) - x_i(g)\right) + F' \times (x_{i2}(g) - x_{i3}(g)) \quad (10)$$

where i_1, i_2 and i_3 are indexes of food source randomly chosen so that $i_1 \neq i_2 \neq i_3 \neq i$. k_{ij} is a random value from a uniform distribution between 0 and 1. $F' \in [0,1]$ a random value from a uniform distribution created once for each iteration.

$$v_{ij} = gbest_j + F \times (x_{dets,j} - x_{src,j}) \quad (11)$$

where $gbest$ denotes the bee with the currently best fitness value. $x_{dest,j}$ and $x_{src,j}$ are the bees chosen randomly such that $f(x_{dest,j}) < f(x_{src,j})$, f being the objective function to minimize. F here is a randomly chosen value from a uniform distribution between 0 and 1. The steps of the hybrid ABC algorithm are given in Algorithm 1.

http://dx.doi.org/10.21622/ACE.2024.04.2.852

Algorithm 1 Hybrid-ABC algorithm.

1: Initialize the population of solutions x_{ij};
2: Initialize gbest using K-means algorithm;
3: Evaluate the fitness value $fit(x_i)$ of solutions x_i using Eq. 7;
4: repeat
5: Produce a new solution using Eq.10;
6: Evaluate the fitness value of the new solution using Eq. 7;
7: Calculate the probabilities using Eq.9;
8: Produce the new solutions (new positions) v_i for the onlookers from the solutions x_i, selected depending on P_i, and evaluate them;
9: Determine the abandoned solution (source), if exists, and replace it with a new randomly produced solution x_i for the scout using the Eq. 6;
10: Memorize the best solution found so far;
11: until stop condition is satisfied
12: return $gbest$;

IV. EXPERIMENTAL RESULTS

In this paper, all simulation experiments are performed on Windows 10 operating system in a computer with an Intel i7 processor and 8 GB of RAM. ABC and the proposed Hybrid-ABC codes are written in Python and run on the PyCharm platform. For both ABC and Hybrid-ABC algorithms, the following parameters are used. The population number, the limit value, and the maximum number of iterations are set as 100, 50, and 50, respectively.

In the simulations, two different datasets are used. The First Dataset is prepared using the ULAKNET dataset from the Topology Zoo website, while the Second Dataset is created by weighting each node of the data in the First Dataset according to their population density ratio. In the generation of datasets, connections between cities (nodes) are used. The number of nodes between both cities is determined to be the shortest by using the Dijkstra algorithm. Determined node values are kept in a matrix in the form of 81 × 81. This matrix is used as the First Dataset. Having a controller in the regions, where the network density is higher in SDN, reduces the delay in the network and increases the efficiency of the network. For this reason, the Second Dataset is created as each node is weighted according to population density. The weighting formula for each node is given as follows,

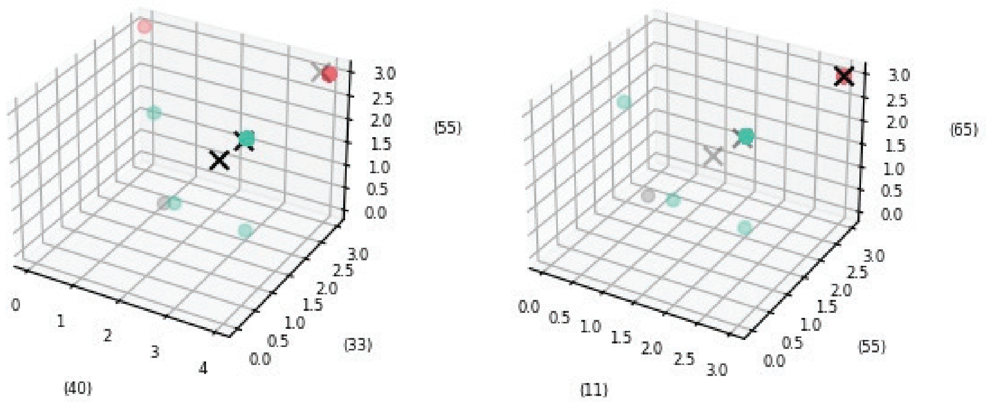
$$^{p}/_{tp} \times n = new\_n \tag{12}$$

where p, tp, n, new_n values represent the population of a city, the total population, the node, and the newly produced node value, respectively. Thus, the First Dataset is primarily focused on network topology derived from the ULAKNET dataset, while the Second Dataset emphasizes population density as a criterion to weight nodes for optimizing SDN. Table 1 demonstrates the main characteristics of the used datasets.

The created datasets are given as inputs to the algorithms. The results obtained from the algorithms are repeated 30 times and the most frequently repeated controller locations are determined. The controller placements obtained for the first run with the first and second datasets as input for ABC and Hybrid-ABC algorithms are shown in Figures 1 and 2, respectively. Besides that, the most frequently repeated controller placements are given in Figures 3 and 4, with the First Dataset and the Second Dataset as input , for ABC and Hybrid-ABC algorithms. In Figures 3 and 4, the orange nodes denote the switches, and the red nodes denote the controllers.

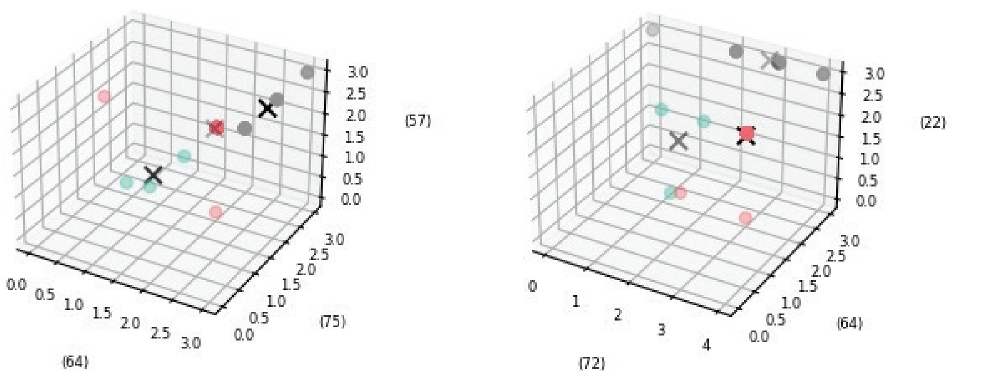
TABLE I
DATASETS CHARACTERISTICS

Datasets	Characteristics
The First Dataset	- Derived from ULAKNET dataset from the Topology Zoo website. - Generated using connections between cities (nodes). - Utilizes the Dijkstra algorithm to determine shortest paths between cities. - Stored as a matrix in an 81 × 81 format.
The Second Dataset	- Shared the same characteristics of the First Dataset. - Created by weighting nodes based on population density ratio. - Intended to optimize SDN by assigning weights to each node.



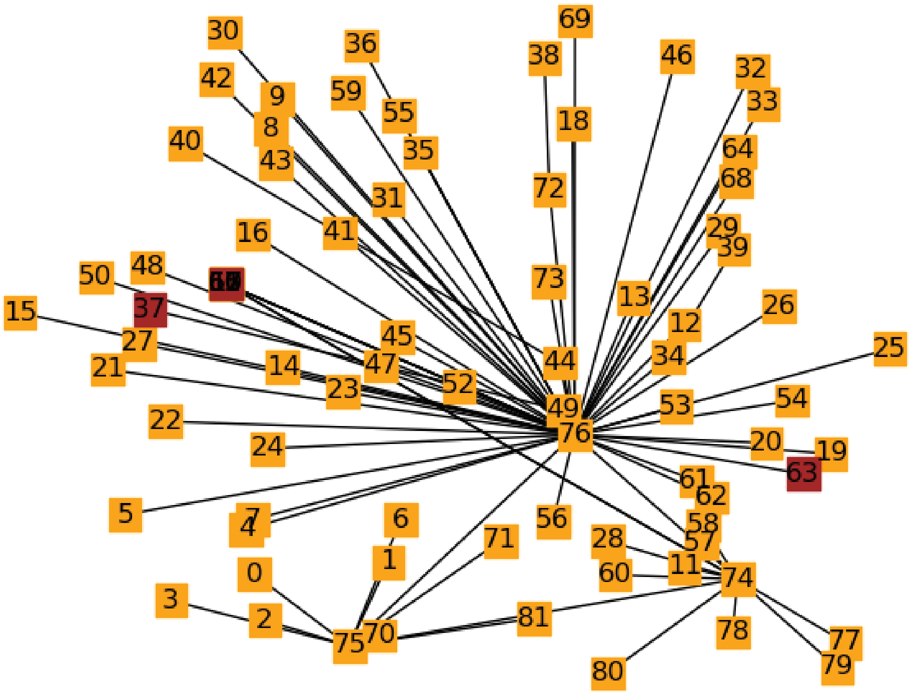
(a) (b)

Fig. 1. Controller placements result for the First Dataset as input after one iteration run according to: (a) ABC, (b) Hybrid-ABC

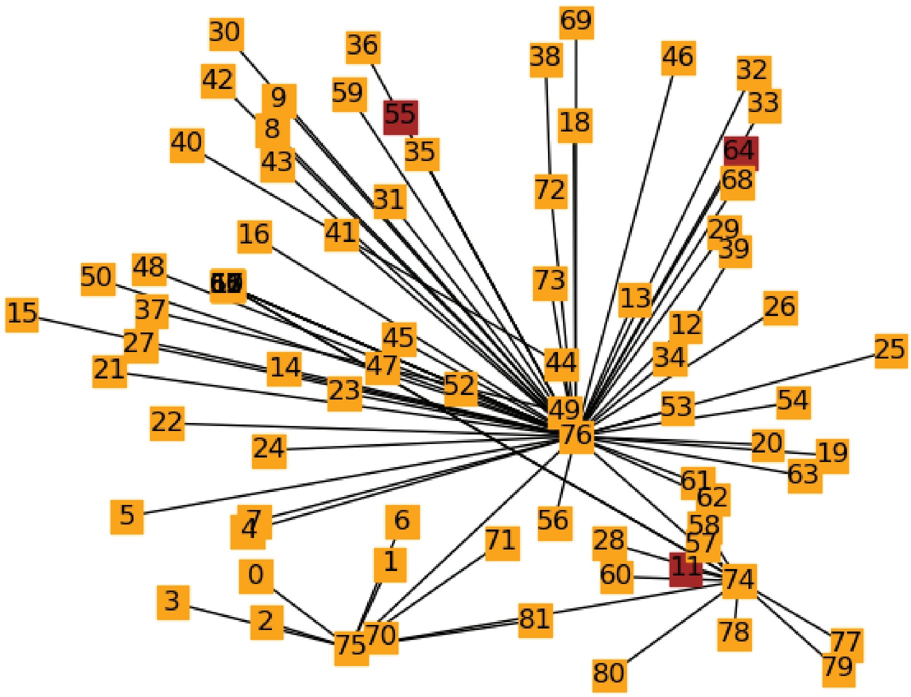


(a) (b)

Fig. 2. Controller placements result for the Second Dataset as input after one iteration run according to: (a) ABC, (b) Hybrid-ABC



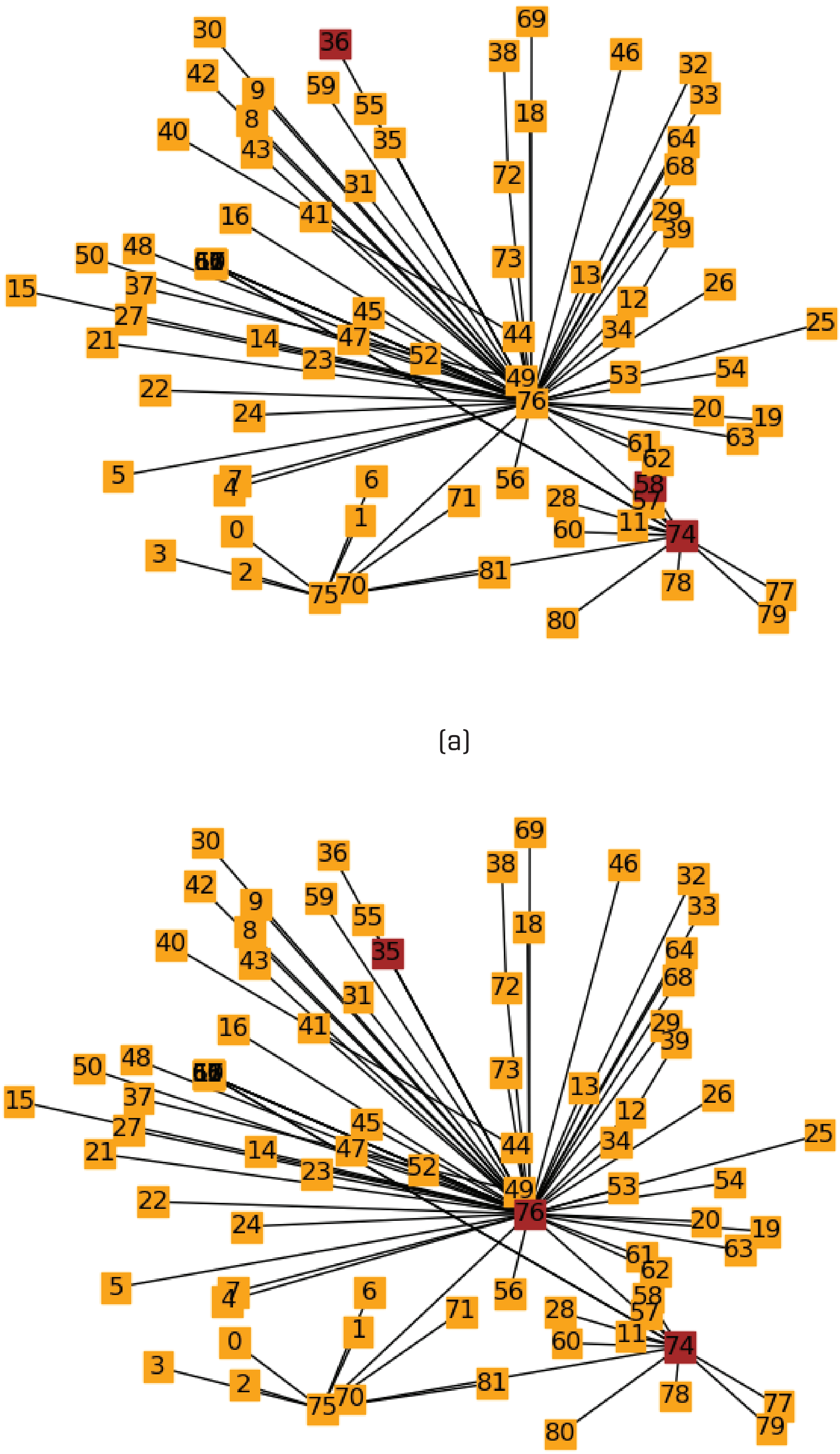
(a)



(b)

Fig. 3. Most repeated controller placements result for the First Dataset as input after one iteration run according to: (a) ABC, (b) Hybrid-ABC

http://dx.doi.org/10.21622/ACE.2024.04.2.852



(a)

(b)

Fig. 4. Most repeated controller placements result for the Second Dataset as input after one iteration run according to: (a) ABC, (b) Hybrid-ABC

http://dx.doi.org/10.21622/ACE.2024.04.2.852

In Figures 3 and 4, 76 represents the first-busiest province, while 74 represents the second-busiest province of Turkey. The controller placement directly affects network performance in SDN; therefore, in Fig. 3, when the results obtained from the First Dataset are compared with the results obtained from the Second Dataset in Fig. 4, it is obvious that the controller placement in the Second Dataset is more efficient in terms of energy-efficiency as the Second Dataset is characterized the population density of each city.

In the controller placement problem, it is aimed to minimize the average delay according to the controller positions and, find approximately optimal solutions in a short time. In Table II, the average latency times obtained as a result of placing the controllers according to both datasets are given milliseconds and the results of ABC and Hybrid-ABC algorithms in terms of quality of solution, standard deviation, and running time are listed in Table III for the first and the second dataset.

TABLE II
COMPARISON OF THE AVERAGE LATENCY TIMES OF THE CONTROLLERS FOR ABC AND HYBRID-ABC ALGORITHMS

Datasets	Algorithms	Controllers Average Latency (ms)
First Dataset	ABC	2.88
	Hybrid-ABC	1.33
Second Dataset	ABC	2.50
	Hybrid-ABC	1.69

In Table II, the controller latency times of ABC and Hybrid-ABC algorithms for two different datasets are compared.

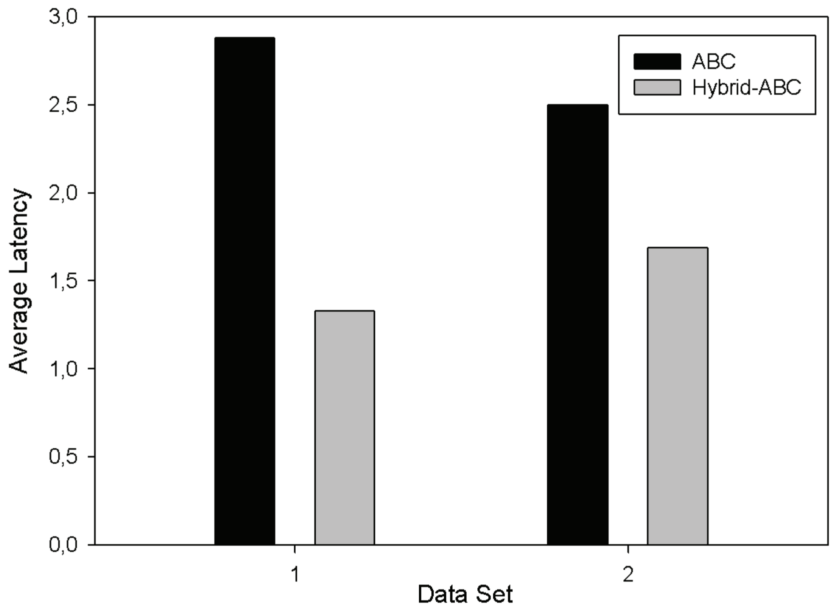


Fig. 5. Average latency results of ABC and Hybrid-ABC for both datasets

http://dx.doi.org/10.21622/ACE.2024.04.2.852

It is clearly seen from Table II and Fig. 5 that the Hybrid-ABC algorithm finds a better location than the ABC algorithm and reduces the latency. This comparison between ABC and Hybrid-ABC algorithms for addressing the multi-CPP in SDN across two datasets, reveals significant improvements in average latency. The Hybrid-ABC algorithm outperforms the traditional ABC approach by reducing latency by approximately 53.82% (2.88 ms to 1.33 ms) and 32.40% (2.50 ms to 1.69 ms) across the First and the Second datasets, respectively. These results indicate that the integration of K-means within the Hybrid-ABC algorithm effectively optimizes controller placement due to the fact that K-means clustering facilitates more effective grouping of network elements which aids Hybrid-ABC in making informed decisions about controller placement based on network characteristics. Consequently, this enabled the proposed algorithm to minimize communication latency between controllers and switches by strategically positioning them.

TABLE III
COMPARISON PERFORMANCE OFABC AND HYBRID-ABC ALGORITHMS FOR THE CPP PROBLEM

Datasets	Algorithms	Quality of Solution	Standard Deviation	Running Time (ms)
First Dataset	ABC	228.78	4.31	242.80
	Hybrid-ABC	223.60	2.98	227.30
Second Dataset	ABC	256.92	4.24	234.98
	Hybrid-ABC	254.96	0.95	233.95

In Table III, the solution quality is measured by the sum of squares error. Table III clearly shows that the performance of the proposed Hybrid-ABC algorithm is better than that of the ABC algorithm. It is observed from Table III that with the use of Hybrid-ABC, the standard deviation is decreased for datasets and the quality of the solutions is augmented. In addition, K-means has a considerable impact on the convergence of the algorithm in the use of Hybrid-ABC, therefore, the running time is decreased for datasets. The enhancements in quality of solution, standard deviation, and running time can be attributed to the integration of K-means clustering with the ABC algorithm. This combination improves the exploitation capabilities of the ABC algorithm, allowing for more efficient and accurate controller placements. For instance, in the first dataset, the Hybrid-ABC algorithm achieves a quality of solution of 223.60 compared to 228.78 with the traditional ABC, a reduction in standard deviation from 4.31 to 2.98, and a decrease in running time from 242.80 ms to 227.30 ms. In the second dataset, the Hybrid-ABC algorithm achieves a quality of solution of 254.96 compared to 256.92 with the traditional ABC, a reduction in standard deviation from 4.24 to 0.95, and a decrease in running time from 234.98 ms to 233.95 ms. The K-means clustering component helps in better initialization and refinement of solutions, leading to a higher quality of solution with lower standard deviation. This reduces the variability in the results and ensures more consistent performance. Additionally, the refined search process results in faster convergence, thereby reducing the running time. However, it is important to note that the integration of K-means clustering may introduce some constraints, such as potential sensitivity to the initial cluster centers and outliers. These constraints will be considered and addressed in future work to further enhance the robustness and efficiency of the Hybrid-ABC algorithm.

Consequently, energy efficiency in SDN is closely related to the optimal placement of controllers. Proper controller placement minimizes the distance between controllers and network nodes, which in turn reduces the amount of data that needs to be transmitted over the network. This reduction in data transmission can lead to lower energy consumption by reducing the load on network links and switches. The researchers' proposed Hybrid-ABC algorithm improves solution quality and reduces latency, as evidenced by the results in Tables II and III. Lower latency and better-quality solutions not only enhance network performance but also contribute to energy efficiency. By optimizing controller placement, the researchers' approach helps in reducing the overall energy required for data transmission and processing. Thus, the benefits of lower latency and improved solution quality indirectly support better energy efficiency.

V. CONCLUSION

In this paper, the researchers' introduced ABC and proposed Hybrid-ABC algorithms for solving the multi-CPP within SDN environments. The proposed Hybrid-ABC algorithm combines machine learning and swarm intelligence approaches, showcasing its potential to deliver successful results in solving the CPP. To evaluate these algorithms, the researchers' constructed two distinct datasets. The initial dataset is formulated by rearranging the ULAKNET data sourced from the Topology Zoo database. Subsequently, we generated a second dataset by applying weights based on population density to the initial data, thereby enhancing energy efficiency within the controller setup and layout. The proposed Hybrid-ABC algorithm is devised by leveraging the K-means algorithm to determine the $gbest$ value, along with integrating diverse neighborhood development strategies into the traditional ABC approach. The researchers' experimental results demonstrated that the proposed Hybrid-ABC approach can reduce controller latency and algorithm running time across both datasets. Therefore, the proposed Hybrid-ABC algorithm exhibits promising potential in enhancing network performance metrics, thereby emphasizing its crucial significance in advancing the efficacy and efficiency of SDNs. Consequently, by enhancing the performance of SDNs, organizations and network operators can access numerous benefits, including increased agility, scalability, cost-effectiveness, adaptability to new technologies, and overall network robustness, which are vital in dynamic and continually evolving networking environments. As a direction for future research, the researchers aim to delve deeper into exploring simpler yet more effective optimization algorithms relevant to the multi-CPP. Additionally, comparative studies between these algorithms and the Hybrid-ABC approach will be conducted to evaluate their respective efficiencies and performance in the CPP within SDN environments.

VI. REFERENCES

[1] R. Amin, E. Rojas, A. Aqdus, S. Ramzan, D. Casillas-Perez, and J. M. Arco, "A Survey on Machine Learning Techniques for Routing Optimization in SDN," 2021. doi: 10.1109/ACCESS.2021.3099092.

[2] D. Espinel Sarmiento, A. Lebre, L. Nussbaum, and A. Chari, "Decentralized SDN Control Plane for a Distributed Cloud-Edge Infrastructure: A Survey," IEEE Communications Surveys and Tutorials, vol. 23, no. 1, 2021, doi: 10.1109/COMST.2021.3050297.

http://dx.doi.org/10.21622/ACE.2024.04.2.852

[3] T. Hu, Z. Guo, P. Yi, T. Baker, and J. Lan, "Multi-controller Based Software-Defined Networking: A Survey," IEEE Access, vol. 6, 2018, doi: 10.1109/ACCESS.2018.2814738.

[4] J. Lu, Z. Zhang, T. Hu, P. Yi, and J. Lan, "A survey of controller placement problem in software-defined networking," IEEE Access, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2893283.

[5] G. Wang, Y. Zhao, J. Huang, Q. Duan, and J. Li, "A K-means-based network partition algorithm for controller placement in software defined network," in 2016 IEEE International Conference on Communications, ICC 2016, 2016. doi: 10.1109/ICC.2016.7511441.

[6] B. BABAYİĞİT and B. ULU, "A High Available Multi-Controller Structure for SDN and Placement of Multi-Controllers of SDN with Optimized K-means Algorithm," Iğdır *Ü*niversitesi Fen Bilimleri Enstitüsü Dergisi, vol. 11, no. 4, 2021, doi: 10.21597/jist.932575.

[7] B. P. R. Killi, E. A. Reddy, and S. V. Rao, "Cooperative game theory based network partitioning for controller placement in SDN," in 2018 10th International Conference on Communication Systems and Networks, COMSNETS 2018, 2018. doi: 10.1109/COMSNETS.2018.8328186.

[8] L. Liao and V. C. M. Leung, "Genetic algorithms with particle swarm optimization based mutation for distributed controller placement in SDNs," in 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks, NFV-SDN 2017, 2017. doi: 10.1109/NFV-SDN.2017.8169836.

[9] K. S. Sahoo, S. Sahoo, A. Sarkar, B. Sahoo, and R. Dash, "On the placement of controllers for designing a wide area software defined networks," in IEEE Region 10 Annual International Conference, Proceedings/TENCON, 2017. doi: 10.1109/TENCON.2017.8228398.

[10] Q. Zhang et al., "A new quantum particle swarm optimization algorithm for controller placement problem in software-defined networking," Computers and Electrical Engineering, vol. 95, 2021, doi: 10.1016/j.compeleceng.2021.107456.

[11] Y. Hu, T. Luo, N. C. Beaulieu, and C. Deng, "The energy-aware controller placement problem in software defined networks," IEEE Communications Letters, vol. 21, no. 4, 2017, doi: 10.1109/LCOMM.2016.2645558.

[12] V. Ahmadi and M. Khorramizadeh, "An adaptive heuristic for multi-objective controller placement in software-defined networks," Computers and Electrical Engineering, vol. 66, 2018, doi: 10.1016/j.compeleceng.2017.12.043.

[13] B. Babayigit, B. Ulu, and E. N. Hascokadar, "Solving Multi-Controller Placement Problem in Software Defined Networks with A Genetic Algorithm," in UBMK 2019 - Proceedings, 4th International Conference on Computer Science and Engineering, 2019. doi: 10.1109/UBMK.2019.8907199.

[14] G. D'Angelo and F. Palmieri, "A co-evolutionary genetic algorithm for robust and balanced controller placement in software-defined networks," Journal of Network and Computer Applications, vol. 212, 2023, doi: 10.1016/j.jnca.2023.103583.

[15] D. He, J. Chen, and X. Qiu, "A density algorithm for controller placement problem in software defined wide area networks," Journal of Supercomputing, vol. 79, no. 5, 2023, doi: 10.1007/s11227-022-04873-x.

[16] M. Dhar, A. Debnath, B. K. Bhattacharyya, M. K. Debbarma, and S. Debbarma, "A comprehensive study of different objectives and solutions of controller placement problem in software-defined networks," Transactions on Emerging Telecommunications Technologies, vol. 33, no. 5, 2022, doi: 10.1002/ett.4440.

[17] S. Guan, J. Li, Y. Li, and Z. Wang, "A multi-controller placement method for software defined network based on improved firefly algorithm," Transactions on Emerging Telecommunications Technologies, vol. 33, no. 7, 2022, doi: 10.1002/ett.4482.

[18] N. Lin, Q. Zhao, L. Zhao, A. Hawbani, L. Liu, and G. Min, "A Novel Cost-Effective Controller Placement Scheme for Software-Defined Vehicular Networks," IEEE Internet Things J, vol. 8, no. 18, 2021, doi: 10.1109/JIOT.2021.3069878.

[19] N. S. Radam, S. T. F. Al-Janabi, and K. S. Jasim, "Using Metaheuristics (SA-MCSDN) Optimized for Multi-Controller Placement in Software-Defined Networking," Future Internet, vol. 15, no. 1, 2023, doi: 10.3390/fi15010039.

[20] S. Malakar, M. Ghosh, S. Bhowmik, R. Sarkar, and M. Nasipuri, "A GA based hierarchical feature selection approach for handwritten word recognition," Neural Comput Appl, vol. 32, no. 7, 2020, doi: 10.1007/s00521-018-3937-8.

[21] N. Bacanin, R. Stoean, M. Zivkovic, A. Petrovic, T. A. Rashid, and T. Bezdan, "Performance of a novel chaotic firefly algorithm with enhanced exploration for tackling global optimization problems: Application for dropout regularization," Mathematics, vol. 9, no. 21, 2021, doi: 10.3390/math9212705.

[22] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," Journal of Global Optimization, vol. 39, no. 3, 2007, doi: 10.1007/s10898-007-9149-x.

[23] M. Abubaker and W. Ashour, "Efficient Data Clustering Algorithms: Improvements over Kmeans," International Journal of Intelligent Systems and Applications, vol. 5, no. 3, 2013, doi: 10.5815/ijisa.2013.03.04.

[24] M. B. A. - and H. M. H. -, "Kmeans-Based Convex Hull Triangulation Clustering Algorithm," Research Notes in Information Science, vol. 9, no. 1, 2012, doi: 10.4156/rnis.vol9.issue1.3.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

OPTIMAL DESIGN OF LAMINATED COMPOSITE AND NANOCOMPOSITE STRUCTURES USING EVOLUTIONARY OPTIMIZATION TECHNIQUES: A SURVEY

Ranganai T. Moyo[1], and Pavel Y. Tabakov[2]

[1] Department of Mechanical Engineering, Durban University of Technology, Durban, 4001, South Africa
[2] Department of Mechanical Engineering and Institute for Systems Science, Durban University of Technology, Durban, 4001, South Africa

Emails: {moyoranganai@gmail.com, pashat@dut.ac.za}

Received on, 16 July 2024 - Accepted on, 24 August 2024 - Published on, 05 September 2024

ABSTRACT

The optimal design of laminated composite and nanocomposite (LCNC) structures stands at the forefront of materials engineering, offering the potential to revolutionize the development of advanced materials with superior mechanical, thermal, and electrical properties. By tailoring LCNC structures to meet specific performance requirements, optimizing material usage, and exploring innovative design approaches, engineers can create lighter, more efficient, and environmentally friendly structures that excel in diverse applications. Many industries such as automotive, aerospace, and construction are already using composite and nanocomposite materials to develop high-strength and lightweight structures. Thus, this survey delves into evolutionary optimization techniques as powerful tools for achieving optimal configurations in LCNC structures, highlighting the importance of selecting the appropriate technique for a given optimization problem. A strict selection method was employed to come up with this review paper, and only reputable literary sources were used. The research articles used in this survey were searched from top research databases such as ScienceDirect, IEEE Xplore, Scopus, and Google Scholar. The articles published in the period, 2015 to 2024 were considered. Common design optimization problems such as buckling load, vibration, and weight and cost minimization were covered.

Index words: LCNC structures; Evolutionary optimization; Optimal design; Advanced materials.

I. INTRODUCTION

Advancements in material technologies have accelerated the use of many new materials in diverse applications. Among the available options, fibre-reinforced composite materials, especially laminated composites, are recognized to have the potential to bring about a future material revolution, especially because of their flexible functionalities and related advantages [1]. Laminated composite materials are well known for their high stiffness, strength, lightweight, longer fatigue life, and corrosion resistance. Their key benefit is that they can be tailored to suit the intended application and for this reason, the demand and growth for these materials are on the rise [2]. More recently, the field of composite design witnessed the

http://dx.doi.org/10.21622/ACE.2024.04.2.930

advent of nanocomposite materials, and these materials have since attracted much attention due to their significant improvement in their mechanical, chemical, and thermal properties. Nanocomposites can be defined as materials that incorporate nanofillers into a matrix of a standard composite material [3]. Figure 1 shows the schematic diagram for the formation of a laminated nanocomposite structure.

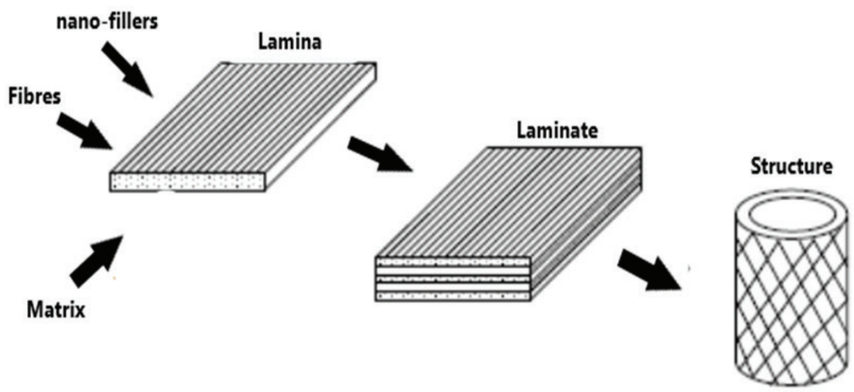


Fig. 1. Schematic diagram showing the formation of laminated nanocomposite structure

Nanofillers in nanocomposites represent a cutting-edge area of materials science where integrating nanoscale materials into a standard composite leads to enhanced properties and novel functionalities. Commonly used nanofillers in composite materials include nano-clay, carbon nanotubes (CNTs), carbon nanofibers (CNFs), and graphene nanoplatelets (GNPs) [4]. Both laminated composite and nanocomposite (LCNC) materials are advanced materials, and they are well-suited to applications that require lightweight and versatile structural designs. Figure 2 demonstrates that LCNC materials are slowly replacing traditional materials as far as weight and cost minimization are concerned.

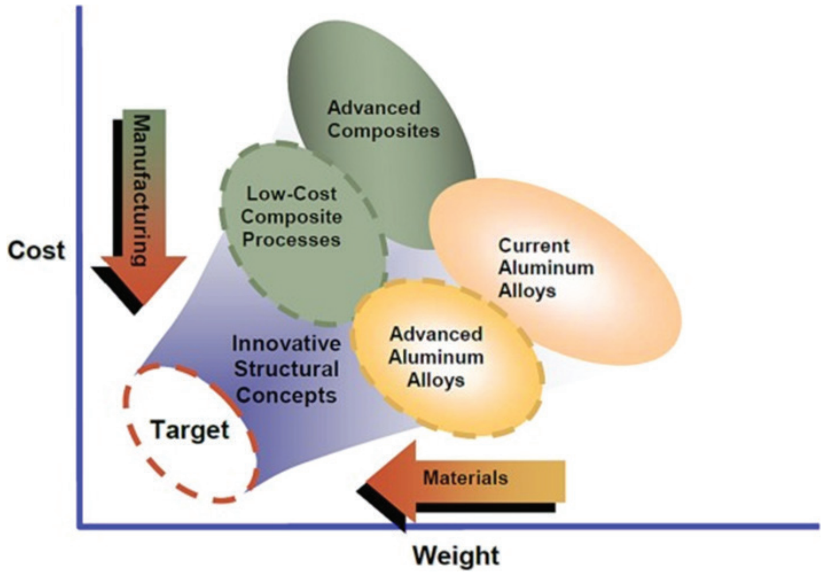


Fig. 2. Weight and cost consideration for different materials selection [5]

Many industries such as automotive, aerospace, and construction are already using LCNC materials to develop high strength and lightweight structures to minimize the cost of materials. However, the complex behavior of these materials creates a need to first validate the performances of the structures before use [6]. Thus, a process of optimization is crucially required when designing LCNC structures. Over several years, numerous techniques have been developed and suggested for the optimal design of LCNC structures. Most of these optimization techniques are experimental, analytical, and classical numerical methods. The designs based on physical experimentation are expensive and consume much time to reach the desired outcome [7]. Analytical and classical numerical optimization methods are limited when it comes to non-linear and multi-objective optimization problems, which is mostly the case when designing LCNC structures [8]. Such methods suffer from the lack of robustness, and they are hampered by inauspicious features in the multi-dimensional space like "ridges", "canyons", "flat spots" and multiple extremes. Besides, they are local in scope; the optima they seek are the best in the neighbourhoods of the current point. With the advent of evolutionary computation, many new optimization algorithms have been developed as an alternative to analytical and classical numerical techniques for the design of LCNC structures. These evolutionary algorithms have since proved to be effective in various applications because of their ability to handle complex, multi-dimensional, and non-linear optimization problems commonly found in structural design. In [9-11], the genetic algorithm (GA) was successfully utilized for the optimal design of different types of LCNC structures. The particle swarm optimization (PSO) technique was also implemented to design LCNC structures in [12, 13]. Some other evolutionary algorithms that have been applied for the design optimization of LCNC structures include differential evolution (DE) [14], simulated annealing (SA) [15], and ant colony optimization (ACO) [16]. These examples and many more confirm the supremacy of evolutionary computation for the optimal design of LCNC structures.

Although evolutionary algorithms prove to be superior to conventional optimization techniques, their applicability depends on the nature of the design problem at hand, and selecting the appropriate algorithm for a given application is of prime importance. Thus, this paper provides a review of the available studies related to the design optimization of LCNC structures using evolutionary algorithms highlighting the importance of selecting the appropriate technique for a given design problem. Moreover, it should be noted that many review papers on this topic, concentrate solely on composite structures without considering nanocomposite structures and this paper aims to fill that gap by exploring the optimization of both LCNC structures. Moreover, the survey is limited to the study of LCNC beams, plates, panels, and shells.

II. COMPOSITE AND NANOCOMPOSITE STRUCTURES

LCNC structures have been useful in various applications such as automotive, aerospace, construction, maritime, and energy. They offer a remarkable strength-to-weight ratio which makes them significantly lighter than traditional materials while maintaining high levels of strength and stiffness, making them advantageous in weight-sensitive applications. The next sections cover different types of LCNC structures which include beams, plates, panels, and shells.

A. BEAMS

Different types of LCNC beams, straight and curved, have been developed and employed in numerous engineering applications such as for constructing bridges and buildings. With superior strength-to-weight ratios, they excel in applications where weight reduction is of prime importance. Figure 3 shows the microscopic geometry of a typical laminated composite beam.

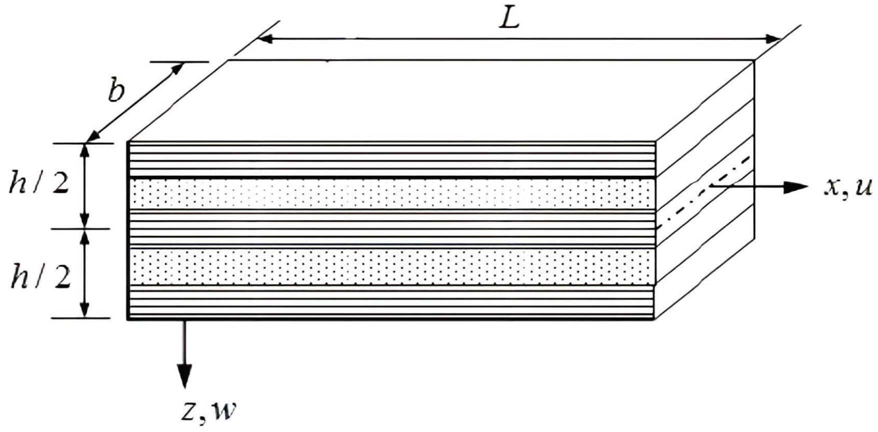


Fig. 3. Typical geometry of a laminated composite beam

In a study by Lotfy et al. [17], composite straight beams were used to develop railroad crossties and they proved to be reliable and durable compared to traditional wooden counterparts. LCNC materials can be also utilized to produce bumper beams. Bumper beams are structural components used in the automotive industry as part of the vehicle's bumper system. The main objective of the bumper system is to absorb the kinetic energy in accidents by deflection or deformation during crashes. Zhu et al. [18] designed and optimized a composite bumper beam with variable cross-sections for automotive vehicles. Their design demonstrated superiority to conventional metallic bumper beams in terms of lightweight and crashworthiness. Another type of LCNC beam is the box beam, which is the most useful structural element used for turbine and aircraft designs. A study by Qin et al. [19] presented the design and nonlinear analysis of a multi-bolted joint composite box-beam for sectional wind turbine blades. Their design performed better than the conventional box-beams studied.

B. PLATES

Plates in composite science are described as structures with one of the dimensions noticeably smaller than the other two [20]. These structures are employed in several industries because of their high strength and lightweight functionalities. Common applications of composite plates include aircraft components (such as wings and fuselage panels), automotive parts (like body panels and structural components), boat hulls, sporting goods (such as tennis rackets and bicycle frames), and civil engineering structures (such as bridges and building facades). The inclusion of nanofillers in composite plates has significantly improved their performance and increased their applications. The most common and used composite plate is the conventional laminated plate. To improve their performance, conventional laminated composite plates have been designed and optimized in several studies. The studies [21-23] showed the vibration analysis and frequency optimization of composite plates, and the buckling analysis was tackled in [24-27] and the articles [28, 29] which explored damage detection on laminated composite plates. Laminated nanocomposite plates were also designed and optimized in [30, 31]. Some others that frequently used composite and nanocomposite plates include perforated plates [32, 33], skew plates [34, 35] and stiffened plates [36, 37].

http://dx.doi.org/10.21622/ACE.2024.04.2.930

C. SHELLS

Composite shells, crafted from materials like fiberglass or carbon fiber embedded in an epoxy resin matrix, represent a versatile solution employed across diverse industries. Renowned for their lightweight nature, composite shells find extensive use in aerospace for components like fuselage sections and wings, in marine applications for boat hulls due to their corrosion resistance, and in automotive sectors for body panels, all contributing to enhanced efficiency and performance. Figure 4 gives the schematic diagram of the composite laminated cylindrical shell.

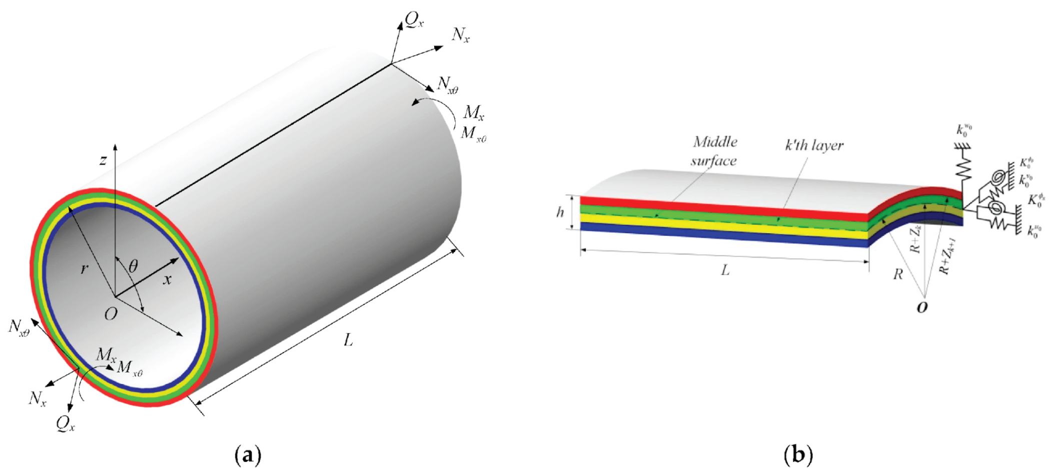


Fig. 4. A schematic diagram of the composite laminated cylindrical shell with elastic boundary conditions (a) whole cylindrical shell (b) the cross-sectional view showing different layers [38]

The addition of nanomaterials in composite shells further improves the versatility and performance of these structures. There are three main types of LCNC shells namely conical, cylindrical, and spherical shells. A typical example of the application of composite cylindrical and spherical shells is manufacturing pressure vessels for storing gases or liquids under high pressure in industries such as oil and gas, chemical processing, and alternative energy. Architectural structures employ composite conical shells for domes and canopies, offering both aesthetic appeal and structural efficiency. The design and optimization of conical shells can be found in studies [39, 40], cylindrical shells were investigated in [39, 41] and spherical shells were analyzed in [42, 43].

D. PANELS

LCNC panels are versatile structural elements typically comprised of a core material sandwiched between two outer layers. With advantages including being light in weight, enhanced strength, and corrosion resistance, LCNC panels find wide applications in construction, manufacturing, and industrial settings. Panels are commonly used for wall cladding, roofing, insulation, partitions, and other architectural and structural elements. Imbalzano et al. [44] designed and studied auxetic composite panels under blast loadings. The designed panels were able to absorb double amount of the impulsive energy. The maximum velocity of the back facet was reduced by 70% as compared to monolithic ones. Nanocomposite panels were optimized in the study by Al-Furjan et al. [45].

 Volume 4, Issue 2, December 2024- ISSN 2735-5985
http://dx.doi.org/10.21622/ACE.2024.04.2.930

III. OVERVIEW OF COMMONLY USED EVOLUTIONARY COMPUTATION TECHNIQUES

Evolutionary computation encompasses various computational paradigms inspired by principles from biological evolution to solve complex problems [46]. These techniques employ algorithms that mimic evolutionary processes like mutation, selection, recombination, and adaptation to generate solutions. Some commonly utilized evolutionary computation methods include:

A. GENETIC ALGORITHM

A genetic algorithm (GA) is a computational optimization technique inspired by the process of natural selection and evolutionary biology. The fundamental theory of the use of the GA to solve problems was first developed by John Holland [47]. Holland pioneered the use of the selection, crossover, and mutation processes and these genetic operators form an essential part of the GA as a problem-solving method. Figure 5 shows the flowchart of a standard GA.

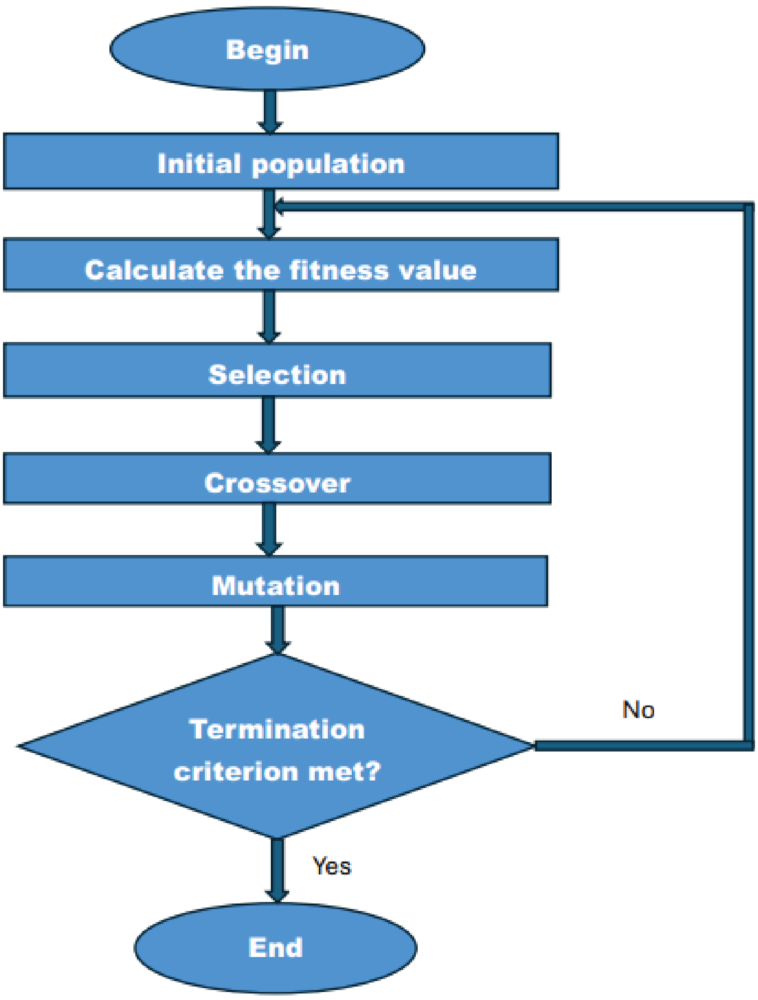


Fig. 5. A flowchart of a standard GA.

Since then, numerous variants of the GA have been developed and implemented in various optimization problems. When implementing the GA, the first step is to select the genes for the initial generation. After that, the best genes in the population

http://dx.doi.org/10.21622/ACE.2024.04.2.930

are extracted through fitness function calculations. Then genes that have the best fitness function values are selected to produce offspring using the crossover and mutation operators. The crossover operator, which is also known as recombination, involves combining the genetic material from selected two parents to create new offspring. The mutation operator involves introducing random changes in the genetic material of individuals within the population. The mutation process helps to maintain genetic diversity and prevents premature convergence of the algorithm. A wide range of GA variants have been developed and employed to solve optimization problems for example the elitism genetic algorithm [48], adaptive genetic algorithms [49], and hybrid genetic algorithms [50] to mention a few. Genetic algorithms offer various advantages including their ability to handle complex, non-linear optimization problems with large solution search spaces, where traditional techniques normally struggle. Their population-based approach promotes diversity, which aids in avoiding local optima and finding globally optimal solutions. Additionally, genetic algorithms are highly adaptable and can accommodate several types of optimization problems, encoding schemes, and fitness functions. However, it should be noted that the performance of a GA highly depends on the choice of parameters such as population size, mutation rate, and crossover rate, which can be challenging to tune effectively. Therefore, the selection of these parameters is of prime importance.

B. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is a metaheuristic optimization technique that exploits the concepts of animals' social behaviors such as birds flocking or fish schooling. It was first developed by Eberhart and Kennedy [51]. Using the flocking analogy, the PSO algorithm maintains a swarm of individuals known as particles, where each particle represents a potential solution. The particles in the swarm have fitness values mapped using the objective function, and each particle has a velocity that determines its direction and range. Each particle adjusts its position and velocity based on its own experience (p_{best}) and the collective experience of the swarm (g_{best}), aiming to converge toward the optimal solution. The PSO algorithm iteratively updates the particles' positions and velocities, influenced by the particles' historical knowledge and the best-performing solutions encountered by the swarm, until a termination condition is met. Several variants of the PSO have been suggested by different researchers and these include generalized particle swarm optimization (GEPSO) [52], pyramid particle swarm optimization (PPSO) [53], and species-based particle swarm optimization [54]. PSO has several advantages, including its simplicity, efficiency, and versatility in solving a wide range of optimization problems. PSO algorithms' straightforward implementation approach and fewer parameters to tune make them accessible to users with varying levels of expertise. Additionally, PSO strikes a balance between exploration and exploitation, robustly navigating through noisy or uncertain objective functions to find optimal or near-optimal solutions. On the other hand, one of the PSO drawbacks include its susceptibility to premature convergence, particularly towards local optima in complex optimization problems, and its sensitivity to parameter settings. Therefore, selecting the best settings is crucial for the optimal performance of this algorithm.

C. DIFFERENTIAL EVOLUTION

Differential evolution (DE) is a population-based algorithm first introduced by Storm and Price [55] for solving complex optimization problems. When implementing the DE algorithm, the first step is to randomly initialize a population of candidate solutions in the search space where each solution is given as a vector of realvalued parameters. After that, the algorithm iteratively improves the population by implementing the mutation operator to generate new candidate solutions. The mutation process is performed by creating a trial vector for all candidate solutions in the population. After the mutation, the crossover operator is applied to combine the trial vector with the original individual, producing potentially improved offspring. These offspring solutions replace their parents in the population based on the selection criterion given by the objective functions of the optimization problem. The mutation and selection stages are repeated until a termination condition is met. It should be noted that, although the DE algorithm and the GA utilize similar operators, the usage of these operators is different. For example, the GA is crossover based and the mutation is only applied to about 1-2% of the population. In the DE algorithm, the mutation operator is applied to each individual while being transferred to the next generation. The crossover operator is not the dominant operator in DE as it is for the GA. The ordinary DE has been modified by several researchers creating new variants such as mixed variable differential evolution (MDE) [56], multiple sub-populations-based DE (MPADE) [57], sinusoidal differential evolution (SinDE) [58], and adaptive guided differential evolution algorithm (AGDE) [59]. The DE algorithm presents several advantages such as its simplicity in implementation, computational efficiency for multi-objective optimization problems, and robustness to parameter settings. Despite being less sensitive to parameter settings compared to other evolutionary algorithms, DE's performance can still be impacted by suboptimal parameter values which can lead to a slower convergence rate.

D. OTHER ALGORITHMS AND HYBRIDIZATION

In addition to the above-mentioned techniques, some other evolutionary algorithms have been successfully implemented for tackling complex optimization problems. One of them is simulated annealing (SA) based on metallurgical annealing. The other one is the grey wolf optimization (GWO) inspired by the leadership and hunting processes of grey wolves. The whale optimization algorithm (WOA), which mimics the hunting mechanism of humpback whales has been also applied to several optimization problems. Some other interesting evolutionary algorithms include the ant colony optimization (ACO), bee colony optimization (BCO), JAYA optimization algorithm, and salp swarm optimization (SSO) algorithm. To improve the performance of evolutionary algorithms some researchers made recourse to hybridization of the algorithms. For example, the PSO can be combined with the GA to give the PSO-GA hybrid algorithm. In most cases, one of the algorithms is used to fine-tune the optimization parameters whereas the other is utilized for the actual optimization process. Machine learning techniques such as artificial neural networks (ANN) can also be combined with evolutionary algorithms to form a hybrid system. For instance, Li et al. [60] combined ANN with the GA (ANN-GA) for the optimization of engine efficiency and emissions. Fuzzy systems such as Fuzzy Logic (FL) can also be hybridized with evolutionary algorithms for optimization [61]. Lastly, traditional, or gradient-based optimization techniques can also be merged with evolutionary algorithms to produce a powerful optimization tool. For example, D'Angelo and Palmieri [62] proposed a hybrid GA that combines the gradient-descent technique with a classical genetic algorithm to solve constrained optimization problems.

IV. APPLICATIONS OF EVOLUTIONARY ALGORITHMS FOR THE OPTIMAL DESIGN OF CNC STRUCTURES

This survey aims to provide an insightful overview of the current state of research, methodologies, and applications of evolutionary algorithms for the optimal design of

http://dx.doi.org/10.21622/ACE.2024.04.2.930

LCNC structures. Evolutionary algorithms have been utilized for the optimal design of composite and nanocomposite structures in various design problems such as buckling load, vibration, and weight and cost minimization. Evolutionary computation techniques present a potent solution for the design of LCNC structures due to their prowess in global optimization, adept handling of non-linearity and multimodality, and capability in multi-objective optimization, enabling designers to balance conflicting objectives effectively.

A strict selection method was employed to come up with this comprehensive review paper. The research articles used in this survey were searched from top research databases such as ScienceDirect, IEEE Xplore, Scopus, and Google Scholar. The articles published in the period, 2015 to 2024 were selected. Additionally, the research articles with the most citations were selected to make sure that the used material is relevant and credible. To narrow down the scope of this survey, the article focuses on the application of evolutionary algorithms to tackle (a) buckling problems (b) vibration problems, and (c) optimum weight and cost problems, for the optimal design of LCNC structures. Additionally, the survey is limited to the study of LCNC beams, plates, panels, and shells.

A. BUCKLING PROBLEMS

In engineering and structural design, it is essential to prevent buckling, as it can lead to catastrophic failure of a component or structure. Buckling optimization involves finding the optimal design parameters, such as dimensions, material properties, and support conditions, to minimize the risk of buckling under given loading conditions. Several evolutionary algorithms have been suggested for the buckling optimization of LCNC structures.

Karakaya and Soykasap [63] proposed a genetic algorithm and a generalized pattern search algorithm for the optimal design of a composite panel. The composite panel was defined as simply supported on four sides and subjected to biaxial in-plane compressive loads. A 64-ply laminate made of graphite/epoxy was considered and the laminate was regarded as symmetric and balanced with discrete fiber angles $0_2, \pm 45, 90_2$ in the laminate sequence. Several combinations of loading conditions and plate aspect ratio were tested to maximize the critical buckling load. The results were compared with other published research work, and it was found that the genetic algorithm is efficient when it comes to problems that require searching of global optima.

In [64], a challenging multimodal optimization design problem of composite panels was tackled by the use of emergent niching particle swarm optimization (PSO). The adopted algorithms in their study comprised of the species-based PSO (SPSO), the fitness Euclidean-distance ratio-based PSO (FER-PSO), the ring topology-based PSO, and the Euclidean distance-based locally informed particle swarm (LIPS) optimizer. The algorithms were applied to a multimodal buckling maximization problem of composite panels. The results showed that the ring topology-based PSO with no niching parameters was more reliable as compared to the other algorithms and these findings can serve as benchmark solutions for future work.

Ghasemi et al. [65] implemented a new multi-step optimization technique to predict the optimal fiber orientation in glass fiber-reinforced polymer (GFRP) composite shells. The proposed method contained a genetic algorithm coupled with an analytical approach. They considered two critical parameters namely, ultimate buckling load

http://dx.doi.org/10.21622/ACE.2024.04.2.930

and weight of the shells. Other factors such as shell thickness, number of layers, and angle were also considered. To evaluate the critical buckling pressure in GFRP specimens, the obtained results were compared with experimental results. It was concluded that the application of the genetic algorithm caused a decrease of 21% and 28% in the optimum local mass of stiffened unsymmetrical angle-ply and unstiffened symmetrical angle-ply laminated composite shells, respectively.

The improved grey wolf optimization (GWO) technique was utilized for studying the dynamic buckling of laminated truncated nanocomposite conical aircraft shells in moisture and temperature environments as well as in magnetic fields [66]. The laminates of the hybrid nanocomposite were made up of a polymer, carbon fibers, and carbon nanotubes (CNT) based on the Halpin-Tsai model. The improved GWO algorithm was used to optimize the instability and frequency of the structure thereby defining the subjective and objective functions. The main contribution of the study was to maximize the inequality and frequency constraints to control the instability. Moisture, the cone semi-vertex angle, and the number of layers were optimized whilst factors like carbon fiber volume content, temperature influence, CNT radius, and magnetic field were also considered. The results proved that the improved GWO algorithm has better capabilities of searching the global optima compared to the traditional GWO. This is because the proposed algorithm was flexible and able to study the optimum conditions of every problem considered.

Ho-Huu et al. [67] proposed a numerical optimization technique comprised of mixed integer and continuous design variables to optimize the design of laminated composite plates under buckling loads. The thickness of layers and fiber orientation angles were taken as design variables. To analyze the buckling behavior of laminated composite plates, the finite element method was employed. The improved differential evolution algorithm called mixed-variable differential evolution (MDE) was then used to solve the optimization problems. The efficacy of the proposed optimization technique was evaluated using numerical examples. The results concluded that the MDE is effective for solving problems that involve maximization of the buckling load factor.

Liu [68] employed the finite element method and NSGA-II genetic algorithm to determine the maximum buckling load capacity of cylindrical composite shells under axial loading. The parameters such as fiber angle, number of layers, and layer thickness were considered. ABACUS software was utilized to perform the finite element analysis of the composite cylindrical shells for determining the buckling load. The NSGA-II genetic algorithm was then used to modify the layout and thickness of the composite layers to optimize the buckling strength and weight of the structure. The genetic algorithm was successfully able to optimize all geometric characteristics studied.

In [69], the buckling behavior of functionally graded nanocomposite beams reinforced with nano clay was investigated. The specimens were first prepared accordingly and the tensile and buckling tests were also conducted. The GA was then utilized to estimate the Young's modulus of functionally graded nanocomposite beams. The first-order shear deformation beam theory was applied to simply supported beams and the Hamilton principle was then used to derive the governing equations. In addition, the effect of the nano clay on the buckling load was also studied. To evaluate the effectiveness of their technique, a comparison of the theoretical analysis and experimental results was conducted. The results demonstrated high accuracy of the GA for the scenarios presented.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

A hybrid optimization technique based on the adaptive Kriging model and an improved particle swarm optimization algorithm (IPSO) was developed to maximize the buckling load of laminated composite plates [70]. The Kriging model was used for directly predicting the buckling load of laminated plates thereby improving the optimization process. The IPSO algorithm was then utilized to maximize the buckling load. Different factors such as the number of layers, aspect ratios, load type (uniaxial and biaxial), and boundary conditions were considered. The results of the proposed method were then compared to the results in the literature, and it was concluded that the proposed technique was able to achieve optimum results with less computational burden.

Moradi et al. [71] presented a strategy for finding the optimal stacking sequence of stiffened laminated composite panels to achieve their maximum buckling load. The ABAQUS software was used to perform the buckling analysis of the composite panels. An algorithm based on the combination of the PSO and GA was then introduced for the optimization process. The traditional PSO technique was enhanced by introducing a new inertia weight of velocity. The efficiency of the proposed modeling method was evaluated by comparing the modeling results with experimental results and published models. The results proved that the proposed optimization procedure has better performance than existing techniques in terms of accuracy and convergence speed.

A method based on artificial neural networks (ANN) and GA for the optimization of composite laminates was presented in [72]. The buckling loads were predicted using ANN and these ANN models proved to be cost-effective and consumed less computational time as compared to other models. To create the data sets for training and testing the ANN models, the finite element analysis (FEA) and the Latin hypercube sampling (LHS) methods were employed. The genetic algorithm was then utilized for the optimization of the stacking sequences and structural dimensions for maximizing the weight of the laminates. The efficiency of the proposed optimization method was tested by comparing it with other machine learning techniques. The proposed optimization method provided satisfactory results for the problems studied. Table I gives summarized applications of evolutionary algorithms in buckling problems.

TABLE I
A SUMMARY OF EVOLUTIONARY ALGORITHMS IN BUCKLING PROBLEMS

Reference	Evolutionary algorithm	Keynotes
[63]	GA	• The GA was found efficient for finding the global optima. The critical buckling loads were maximized for different plate aspect ratios.
[64]	PSO	• The ring topology-based PSO without any niching parameter was found to be the best compared to other PSO variants for buckling optimization of composite plates.
[65]	GA	• A regenerated GA coupled with an analytical approach was implemented for the ultimate buckling load. The results presented a maximum decrease of 28% in the optimum local mass of the laminates.

[66]	GWO	• The improved GWO was utilized for dynamic buckling optimization of laminated shells. It provided better abilities compared to the standard GWO for searching the global optima.
[67]	MDE	• The MDE was proposed for the numerical design optimization of laminated composite plates. It was able to effectively deal with both the integer and continuous design variables and provided better results compared to a standard DE algorithm.
[68]	NSGA-II	• The NSGA-II was used to find the maximum buckling load of composite shells. The algorithm was able to optimize all the geometric configurations studied.
[69]	GA	• The GA was used for studying the buckling behavior of composite beams. A high accuracy percentage was achieved in identifying Young's modulus after comparing it with experimental results.
[70]	IPSO	• The A-Kriging-IPSO method was utilized to maximize the buckling load of laminated composite plates. The obtained results were in good agreement with the literature results and with minimum computational problems.
[71]	PSO-GA	• A hybrid PSO-GA was proposed for maximum buckling optimization of composite panels. The proposed method had better performance compared to previously published models.
[72]	GA	• The GA was used to predict the buckling load of composite laminates. The proposed method was found efficient for the scenarios studied.

B. VIBRATION PROBLEMS

Understanding the vibrational behavior of composite and nanocomposite structures helps engineers detect potential weaknesses or damage, optimize the designs, predict fatigue life, and ultimately reduce costs by preventing rework and failures. By analyzing natural frequencies, mode shapes, and damping ratios, engineers can tailor LCNC structures to meet operational requirements while maximizing safety, reliability, and efficiency in their applications.

Savran and Aydin [73] employed the differential evolution, Nelder Mead method, and simulated annealing algorithms were utilized for optimizing the fundamental frequency, buckling resistance, and weight of hybrid graphite/epoxy-sitka spruce and graphite-flax/epoxy laminated composite plates. The main motivation of their research was to test the usage of the Sitka spruce as an alternative to synthetic E-glass and natural flax fiber. The authors considered single-objective and multi-objective approaches to acquire optimum design for the hybrid and non-hybrid structures. Then evolutionary algorithms were employed to solve the optimization design problem. Their results showed that the plates designed from hybrid graphite/epoxy-sitka spruce were lighter and had higher frequency and buckling resistance as compared to those designed from glass/epoxy, flax/epoxy, and hybrid graphite-flax/epoxy.

Vosoughi et al. [74] maximized the fundamental frequency of thick laminate with respect to fiber orientations. The governing equations were based on higher-order shear deformation theory and the optimization involved a combination of PSO and genetic algorithms. In the numerical results, authors investigated the effect of problem parameters on the optimal design including layer numbers, thickness ratio, and boundary conditions employing the finite element method for numerical solutions of the problems. Their method proved to be robust and accurate for the scenarios studied.

The nondominated sorting genetic algorithm II (NSGA-II) was suggested by Vo-Duy et al. [11] for the multi-objective optimization of laminated composite beam structures. The main objective was to maximize the natural frequency whilst minimizing the weight of the beam. Fiber volume fractions, fiber orientation angles, and thickness are considered as the design variables. The beam was subjected to a constraint where the natural frequency should be equal to or greater than a predetermined frequency. To carry out free vibrational analysis, the finite element (FE) method with a two-node Bernoulli-Euler beam element was used. The NSGA-II was then employed to solve the multi-objective optimization problem. After comparing their results with published results in the literature, their approach proved to be reliable and effective.

Kalita et al. [75] used a combination of optimization algorithms, namely, GA, repulsive particle swarm optimization with local search and chaotic perturbation (RPSOLS), and co-evolutionary host-parasite (CHP) algorithm for designing skew laminates under varying operational conditions. The FE approach based on the first-order deformation theory was used to measure the natural frequencies of the composite panels. Their research work was based on maximizing the first two modes of the natural frequency by optimizing the stacking sequence of the composite laminates. Their results concluded that the CHP algorithm outperformed both the GA and RPSOLS algorithms in terms of computational speed, accuracy, and reliability.

A global optimization framework based on GA and FE methods was developed by Pal et al. [76] to optimize frequency and avoid resonance in composite shells. For objective function formulation, the first-order shear deformation theory was used. The ply angles were taken as the design variables where only discrete ply angles with 5^0 increments in the ± 90 design space were considered. Various numerical studies were conducted to validate the proposed approach and rectangular and cylindrical shell panels of different radius of curvature and boundary conditions were considered. The obtained results showed an excellent agreement with the published results in the literature.

Peng et al. [77] proposed an optimization procedure that used a combination of GA and PSO to obtain the resonance frequency of the cylindrical composite structure. The nonlocal strain gradient theory (NSGT), zigzag theory, and Hamilton's principle were used for the shear deformable shell displacement field problem formulation. The modal equations of motion were solved by the general differential quadrature element method (GDQEM). The GA-PSO was then utilized to optimize the frequency by varying the angle of composite layers. Their results revealed that after reaching a certain number of layers composite layers (30), increasing the number of layers had no effect on the resonance frequency of the cylinder for all the boundary conditions considered.

Le-Anh et al. [78] proposed a numerical method for the static and fundamental frequency optimization of folded laminated composite plates using an adjusted differential evolution (aDE) algorithm. In their optimization procedure, the objective function was to maximize the fundamental frequency and minimize the strain energy. The fiber orientation was considered as the design variable. For the analysis of the behavior of the folded laminated composite plates, the cell-based smoothed

http://dx.doi.org/10.21622/ACE.2024.04.2.930

discrete shear gap method (CS-DSG3) was used. To search for the optimal solutions, the aDE was utilized. The aDE was designed by integrating the conventional DE for handling discrete integer variables. To check the validity of their study, they compared their results and the results of other algorithms presented in the literature such as GA and PSO. Their optimization procedure proved to be efficient, especially for integer-based design variables.

A multi-objective optimization procedure based on the NSGA-II was proposed to provide optimal natural frequency and cost for graphene/fiber-reinforced nanocomposite laminates [8]. The vibrational analysis was solved by the FE method and the first-order shear deformation theory. To get the effective material properties, micromechanical equations were utilized. The NSGA-II was then used to search for the optimal solutions for different scenarios. For the optimization, four types of design variables which include fiber angles, graphene and fiber distribution, and layer thickness were considered. In their results, they deduced that increasing the graphene nanoplatelets content whilst minimizing fiber content gives cost-effective designs.

Vo-Duy et al. [79] presented a numerical optimization technique for maximizing the fundamental frequency of laminated functional graded carbon nanotube-reinforced composite quadrilateral plates. Their approach was a combination of the CS-DSG3 method for first frequency analysis and the adaptive elitist differential evolution algorithm (aeDE), for solving the optimization problem. The carbon nanotube orientation in the layers was considered as the design variable. In their findings, the obtained frequencies were better than those from existing studies.

Duc et al. [80] proposed a non-linear vibration analysis of laminated composite cylindrical shells using meta-heuristic optimization algorithms. The equations for the analysis were given by the classical shell theory combined with von Karman's nonlinearity and those equations were solved using the Galerkin method. Their optimization procedure involved the minimization of the vibration amplitude and the maximization of the natural frequency. The PSO algorithm and the whale optimization algorithm (WOA) were implemented to solve the optimization problem. Their results showed that the WOA yielded more suitable results for three- and four-layer cases whilst the PSO algorithm performed better for five-layer cases.

An optimization technique for vibration analysis of rotating cross-ply laminated nanocomposite blades was proposed by Xiang et al. [81]. In their study, the vibration analysis and lay-up optimization were done on rotating pre-twisted laminated functional graded CNTs reinforced composite shallow conical shells. The governing equations of the rotating blades were derived and solved by the kp-Ritz method. The GA was then employed to optimize the nondimensional frequency parameter for searching the optimal layering sequence of the rotating shallow conical shells. Their results demonstrated that the GA is a useful tool for sequence optimization of rotating blades in terms of enhancing the vibration characteristics. Table II gives summarized applications of evolutionary algorithms in vibration problems.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

TABLE II
A SUMMARY OF EVOLUTIONARY ALGORITHMS IN VIBRATION PROBLEMS

Reference	Evolutionary algorithm	Keynotes
[73]	DE	• DE was utilized to find the fundamental frequency of laminated composite plates. The algorithm was able to obtain a lightweight structure with a higher frequency & buckling resistance.
[74]	PSO-GA	• A PSO-GA method was used to obtain the maximum fundamental frequency of thick laminated composite plates. The applicability and usability of the proposed technique were tested by solving different problems and the results were satisfactory.
[75]	CHP	• GA, RPSOLC, and CHP algorithms were compared for natural frequency optimization in the design of skew composite laminates. The CHP algorithm outperformed other algorithms in terms of accuracy and computational speed.
[76]	GA	• The GA was employed for the fundamental frequency optimization of composite shells. The GA approach proved reliable compared to other algorithms such as sine cosine algorithms, salp swarm optimization, and ant lion optimizer.
[77]	GA-PSO	• A GA-PSO optimization technique was used to obtain the optimum frequency of laminated rotary nanostructures. The method was able to determine stacking sequences that give the highest frequencies.
[78]	aDE	• The aDE algorithm was proposed for static and frequency optimization of laminated composite plates. The proposed technique outperformed the GA and PSO in terms of reliability and effectiveness.
[8]		• NSGA-II was utilized to find the maximum fundamental frequency of graphene/fiber-reinforced nanocomposite laminates. By employing this technique, an increase in fundamental frequency was noted for the laminates studied.
[79]	aeDE	• The aeDE algorithm was used for the fundamental frequency maximization of laminated composite plates. The algorithm was able to maximize any desirable higher-order frequencies.
[80]	PSO-WOA	• The PSO and WOA algorithms were compared for maximization of natural frequencies of laminated composite shells. The results confirmed that the WOA gave suitable results for cases with 3 & 4 laminate layers while the PSO performed better in five or more-layer cases.
[81]	GA	• The GA was proposed for the optimization of the nondimensional frequency of laminated composite blades. The results proved that the GA is useful for multi-objective optimization problems.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

C. OPTIMUM WEIGHT AND COST PROBLEMS

LCNC materials are essential in modern engineering. However, their cost can be very high, and many efforts must be made to reduce it. Therefore, weight and cost optimization in LCNC materials is crucial for enhancing performance, reducing costs, improving efficiency, and ultimately maintaining competitiveness in the marketplace.

De Munck et al. [82] optimized hybrid composite concrete beams for minimization of both the cost and mass. For finding the optimal solutions, the NSGA-II coupled with a meta-model was utilized. Their optimization procedure gave insights into the influence of different parameters such as the concrete class and span on the cost and weight of composite-concrete beams.

Composite sandwich panels with honeycomb core structures were optimized by Gholami et al. [83] using the PSO method. The panels were subjected to a uniformly distributed load and the Navier-type solution was used to predict the deflection of the panels. To optimize the weight of the panels, the niching memetic PSO (NMPSO) and locally informed particle swarm (LIPS) variants of PSO were implemented. Their results confirmed the effectiveness of the NMPSO in finding optimal solutions for constrained and unconstrained objective functions.

Shrivastava et al. [84] employed a classical GA interfaced with a CAE solver for weight minimization of a carbon fiber composite wing torsion box. Their main objective was minimizing the structural weight. The optimization procedure was characterized by an intelligent laminate selection which was based on static strength, ply orientations, and thickness of the laminates. Their results showed a 29% weight reduction on the aircraft wing torsion box and a 54% reduction in terms of the metallic structure.

Albanesi et al. [85] presented a meta-modal optimization procedure for designing the composite laminate of wind turbine blades. Their methodology combined GA and artificial neural networks (ANN) to minimize the mass and computational cost of the optimization procedure. The proposed method was applied to redesign a 40-kW wind turbine blade to minimize its mass while the structural and manufacturing constraints are fulfilled. The obtained results showed a mass reduction of 20% as well as a 40% reduction in computational cost as compared to the reference design. Table III gives summarized applications of evolutionary algorithms in weight and cost problems.

TABLE III
A SUMMARY OF EVOLUTIONARY ALGORITHMS IN WEIGHT AND COST PROBLEMS

Reference	Evolutionary algorithm	Keynotes
[82]	NSGA-II	• The NSGA-II was utilized for weight and cost minimization of composite beams. The optimization algorithm was able to show the dominance of certain design variables in the design space.
[83]	NMPSO	• Several variants of PSO were employed for the minimization of weight in the design of composite sandwich panels. The results confirmed the superiority and effectiveness of the NMPSO algorithm for finding optimal solutions.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

[84]	GA	• The classical GA was utilized for weight minimization of a carbon fiber composite wing torsion box. The results showed that the proposed technique achieved 54% weight reduction concerning the metallic structure.
[85]	GA	• A GA was used for mass minimization in the design of composite laminated wind turbine blades. The proposed procedure was able to save up to 20% of mass compared to the reference design.

V. CHALLENGES AND FUTURE DIRECTIONS

Navigating the intricate design landscapes inherent in LCNC structures represents a formidable challenge for researchers in the domain of evolutionary optimization. The multidimensional nature of these design spaces, influenced by a myriad of parameters including material composition, fiber orientation, and interface properties, demands sophisticated optimization methodologies capable of efficiently exploring and exploiting such complexities. Moreover, the uncertainty in material properties, stemming from factors such as manufacturing variability and environmental degradation, adds another layer of complexity to the optimization process. Addressing these challenges requires the integration of robustness and reliability considerations into the optimization framework to ensure the validity and applicability of the optimized designs under real-world conditions. In addition to addressing these challenges, future research directions in the field of optimal design of CNC structures using evolutionary optimization techniques hold immense promise for advancing the state-of-the-art. One avenue of exploration involves leveraging machine learning and surrogate modeling techniques to accelerate the optimization process by constructing efficient approximations of complex material behaviors. By training neural networks, these surrogate models can capture the underlying relationships between design parameters and performance metrics, allowing for rapid evaluation of candidate designs without the need for computationally expensive simulations. Furthermore, adaptive and hybrid optimization algorithms offer exciting opportunities for enhancing the convergence speed and solution quality of evolutionary optimization techniques. Adaptive algorithms dynamically adjust their parameters and operators based on problem characteristics and solution progress, allowing for more efficient exploration of the design space and faster convergence to optimal solutions. Hybridizing evolutionary algorithms with other optimization techniques, such as gradient-based methods or local search heuristics, can further exploit their complementary strengths, leading to more robust and versatile optimization frameworks.

Moreover, integrating manufacturing constraints into the optimization process early on is crucial for ensuring the practical feasibility and cost-effectiveness of the optimized designs. By incorporating process simulation models and design for manufacturability (DFM) principles into the optimization framework, researchers can account for manufacturing constraints such as material availability, processing limitations, and production costs during the design optimization process. This holistic approach not only ensures that the optimized designs meet performance requirements but also facilitates a seamless transition from design to manufacturing, thereby reducing time-to-market and enhancing product competitiveness. Additionally, advancing material characterization techniques, such as in-situ microscopy, spectroscopy, and imaging, hold tremendous potential for gaining deeper insights into the structure-property relationships of composite and nanocomposite materials. By clarifying the complex interactions between constituent materials at

http://dx.doi.org/10.21622/ACE.2024.04.2.930

the micro- and nano-scale, these advanced characterization techniques enable the development of more accurate material models and optimization objectives, leading to the design of superior-performing structures with tailored properties.

Furthermore, extending topology optimization techniques to the design of nanocomposite structures represents a promising avenue for unlocking novel material architectures with unparalleled performance characteristics. Topology optimization, which seeks to determine the optimal distribution of material within a given design domain to achieve specified performance objectives, has traditionally been applied to macroscopic structures composed of homogeneous materials. However, by extending this approach to the micro- and nano-scale, researchers can explore a vast array of material architectures and morphologies at the atomic and molecular levels, leading to the discovery of innovative designs with enhanced mechanical, thermal, and electrical properties. By tailoring the distribution and arrangement of nanoparticles or nanofillers within the polymer matrix, researchers can engineer nanocomposite materials with unprecedented combinations of strength, stiffness, toughness, and conductivity, opening new possibilities for applications in aerospace, automotive, biomedical, and renewable energy sectors.

VI. CONCLUDING REMARKS

The authors reviewed over 300 articles, and they did not seek to cite as many sources as possible but adhered to the principle of less but better. The main emphasis was to provide relevant achievements on the application of evolutionary algorithms for the optimal design of LCNC structures. In general, the design optimization of LCNC structures involves systematically improving the performance and efficiency of these materials through exploration and refinement of their design parameters. This process typically begins with defining the objectives (e.g., minimizing weight, maximizing stiffness) and constraints (e.g., material properties, manufacturing limitations) of the structure. Next, the LCNC structure is represented using suitable mathematical models, often involving discretization of design variables such as ply orientations, thicknesses, and stacking sequences. Various optimization techniques are then employed to search for the optimum combination of the design parameters. Thus, through iterative analysis and evaluation, the optimal design structure is identified, balancing performance requirements with practical constraints.

In general, most research work focuses on the utilization of evolutionary algorithms for the design optimization of composite structures, and just a few studies were conducted on nanocomposite structures. It was also found that the GA, particularly the NSGA-II was the most common and efficient evolutionary algorithm for buckling optimization. Different variants of PSO were also considered useful for tackling buckling problems. For vibration analysis and frequency optimization of LCNC structures, three optimization techniques, namely, the GA, PSO, and DE were the best. Hybrid optimization techniques such as GA-PSO were also found to be instrumental in tackling vibration problems. For weight and cost optimization problems, it was found that there were few studies where weight or cost was used for objective function formulation, it was rather considered as a constraint in most cases. Again, the GA was the most common and effective evolutionary algorithm to solve weight and cost optimization problems.

The conclusion that can be drawn from this work is that evolutionary optimization techniques are powerful tools for achieving optimal configuration in LCNC structures. They can easily handle non-linear and multi-objective optimization problems, enabling

http://dx.doi.org/10.21622/ACE.2024.04.2.930

designers to balance conflicting objectives effectively. The best evolutionary algorithms for the optimal design of LCNC structures were found to be the GA and PSO. However, the development and optimization of nanocomposite structures is still lacking, and a lot of work should be done on utilizing these materials. Lastly, this survey should serve as a reference for the researchers interested in studying the application of evolutionary computation for design optimization of LCNC structures.

FUNDING

This research work was funded by the National Research Foundation (NRF) of South Africa, grant number, PMDS2205046450.

CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

LIST OF ABBREVIATIONS

LCNC	Laminated Composite and Nanocomposite
CNTs	Carbon Nanotubes
CNFs	Carbon Nanofibers
GNPs	Graphene Nanoplatelets
GA	Genetic Algorithm
PSO	Particle Swarm Optimization
DE	Differential Evolution
SA	Simulated Annealing
ACO	Ant Colony Optimization
GEPSO	Generalized Particle Swarm Optimization
PPSO	Pyramid Particle Swarm Optimization
MDE	Mixed Variable Differential Evolution
MPADE	Multiple Sub-Population-Based Differential Evolution
SinDE	Sinusoidal Differential Evolution
AGDE	Adaptive Guided Differential Evolution
GWO	Grey Wolf Optimization
WOA	Whale Optimization Algorithm
BCO	Bee Colony Optimization
SSO	Salp Swarm Optimization
ANN	Artificial Neural Networks
FL	Fuzzy Logic

http://dx.doi.org/10.21622/ACE.2024.04.2.930

SPSO	Species-Based Particle Swarm Optimization
FER-PSO	Fitness Euclidean-Distance-Ratio-Based Particle Swarm Optimization.
LIPS	Locally Informed Particle Swarm Optimizer
GFRP	Glass Fiber-Reinforced Polymer
NSGA-II	Non-Dominated Sorting Genetic Algorithm-II
IPSO	Improved Particle Swarm Optimization
FEA	Finite Element Analysis
LHS	Latin Hypercube Sampling
RPSOLS	Repulsive Particle Swarm Optimization with Local Search and Chaotic Perturbation
CHP	Co-Evolutionary Host-Parasite
NSGT	Nonlocal Strain Gradient Theory
GDQEM	General Differential Quadrature Element Method
aDE	Adjusted Differential Evolution
CS-DSG3	Cell-Based Smoothed Discrete Shear Gap Method
aeDE	Adaptive Elitist Differential Evolution
NMPSO	Niching Memetic Particle Swarm Optimization

VII. REFERENCES

[1] J. J. Andrew and H. N. Dhakal, "Sustainable biobased composites for advanced applications: recent trends and future opportunities – A critical review," 2022. doi: 10.1016/j.jcomc.2021.100220.

[2] S. Zhao, Z. Zhao, Z. Yang, L. L. Ke, S. Kitipornchai, and J. Yang, "Functionally graded graphene reinforced composite structures: A review," 2020. doi: 10.1016/j.engstruct.2020.110339.

[3] J. Njuguna, F. Ansari, S. Sachse, V. M. Rodriguez, S. Siqqique, and H. Zhu, "Nanomaterials, nanofillers, and nanocomposites: types and properties," in Health and Environmental Safety of Nanomaterials: Polymer Nanocomposites and Other Materials Containing Nanoparticles, 2021. doi: 10.1016/B978-0-12-820505-1.00011-0.

[4] R. Olayil, V. Arumugaprabu, O. Das, and W. A. Lenin Anselm, "A Brief Review on Effect of Nano fillers on Performance of Composites," in IOP Conference Series: Materials Science and Engineering, 2021. doi: 10.1088/1757-899X/1059/1/012006.

[5] R. Soni, R. Verma, R. Kumar Garg, and V. Sharma, "A critical review of recent advances in the aerospace materials," Mater Today Proc, 2023, doi: 10.1016/j.matpr.2023.08.108.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

[6] Z. K. Awad, T. Aravinthan, Y. Zhuge, and F. Gonzalez, "A review of optimization techniques used in the design of fibre composite structures for civil engineering applications," Mater Des, vol. 33, no. 1, 2012, doi: 10.1016/j.matdes.2011.04.061.

[7] M. Rabothata, J. Muthu, and L. Wegner, "Optimum design parameters and mechanical properties of polymeric nanocomposites using NSGA-II optimization method," J Compos Mater, vol. 55, no. 7, 2021, doi: 10.1177/0021998320960774.

[8] G. A. Drosopoulos, C. Gogos, and G. Foutsitzi, "Multi-objective optimization for maximum fundamental frequency and minimum cost of hybrid graphene/fibre-reinforced nanocomposite laminates," Structures, vol. 54, 2023, doi: 10.1016/j.istruc.2023.05.118.

[9] L. Wang, A. Kolios, T. Nishino, P. L. Delafin, and T. Bird, "Structural optimisation of vertical-axis wind turbine composite blades based on finite element analysis and genetic algorithm," Compos Struct, vol. 153, 2016, doi: 10.1016/j.compstruct.2016.06.003.

[10] Z. Z. Wang and A. Sobey, "A comparative review between Genetic Algorithm use in composite optimisation and the state-of-the-art in evolutionary computation," 2020. doi: 10.1016/j.compstruct.2019.111739.

[11] T. Vo-Duy, D. Duong-Gia, V. Ho-Huu, H. C. Vu-Do, and T. Nguyen-Thoi, "Multi-objective optimization of laminated composite beam structures using NSGA-II algorithm," Compos Struct, vol. 168, 2017, doi: 10.1016/j.compstruct.2017.02.038.

[12] Z. Liu, J. Lu, and P. Zhu, "Lightweight design of automotive composite bumper system using modified particle swarm optimizer," Compos Struct, vol. 140, 2016, doi: 10.1016/j.compstruct.2015.12.031.

[13] H. Li and K. Chandrashekhara, "Particle swarm-based structural optimization of laminated composite hydrokinetic turbine blades," Engineering Optimization, vol. 47, no. 9, 2015, doi: 10.1080/0305215X.2014.954567.

[14] S. Das and P. N. Suganthan, "Differential Evolution: A Survey of the State-of-the-Art," IEEE Transactions on Evolutionary Computation, vol. 15, no. 1, pp. 4–31, Feb. 2011, doi: 10.1109/TEVC.2010.2059031.

[15] D. Delahaye, S. Chaimatanan, and M. Mongeau, "Simulated annealing: From basics to applications," in International Series in Operations Research and Management Science, vol. 272, 2019. doi: 10.1007/978-3-319-91086-4_1.

[16] M. Dorigo and T. Stützle, "Ant colony optimization: Overview and recent advances," in International Series in Operations Research and Management Science, vol. 272, 2019. doi: 10.1007/978-3-319-91086-4_10.

[17] I. Lotfy, M. Farhat, M. A. Issa, and M. Al-Obaidi, "Flexural behavior of high-density polyethylene railroad crossties," Proc Inst Mech Eng F J Rail Rapid Transit, vol. 230, no. 3, 2016, doi: 10.1177/0954409714565655.

[18] G. Zhu, Z. Wang, A. Cheng, and G. Li, "Design optimisation of composite bumper beam with variable cross-sections for automotive vehicle," International Journal of Crashworthiness, vol. 22, no. 4, 2017, doi: 10.1080/13588265.2016.1267552.

[19] Z. Qin, J. Wang, K. Yang, G. Yu, Y. Xu, and J. Xu, "Design and nonlinear structural responses of multi-bolted joint composite box-beam for sectional wind turbine blades," Compos Struct, vol. 206, 2018, doi: 10.1016/j.compstruct.2018.08.073.

[20] C. Y. Lee and W. Yu, "Homogenization and dimensional reduction of composite plates with in-plane heterogeneity," Int J Solids Struct, vol. 48, no. 10, 2011, doi: 10.1016/j.ijsolstr.2011.01.032.

[21] P. Pushparaj and B. Suresha, "Free vibration analysis of laminated composite plates using finite element method," Polymers and Polymer Composites, vol. 24, no. 7, 2016, doi: 10.1177/096739111602400712.

[22] A. Houmat, "Three-dimensional free vibration analysis of variable stiffness laminated composite rectangular plates," Compos Struct, vol. 194, 2018, doi: 10.1016/j.compstruct.2018.04.028.

[23] J. L. Mantari and M. Ore, "Free vibration of single and sandwich laminated composite plates by using a simplified FSDT," Compos Struct, vol. 132, 2015, doi: 10.1016/j.compstruct.2015.06.035.

[24] R. L. Wankhade and S. B. Niyogi, "Buckling analysis of symmetric laminated composite plates for various thickness ratios and modes," Innovative Infrastructure Solutions, vol. 5, no. 3, 2020, doi: 10.1007/s41062-020-00317-8.

[25] K. Shiva, P. Raghu, A. Rajagopal, and J. N. Reddy, "Nonlocal buckling analysis of laminated composite plates considering surface stress effects," Compos Struct, vol. 226, 2019, doi: 10.1016/j.compstruct.2019.111216.

[26] Osman Yassin Osman and Suleiman Osama Mohammed Elmardi, "Buckling Analysis of Thin Laminated Composite Plates using Finite Element Method," International Journal of Engineering Research and Advanced Technology (IJERAT), vol. 3, no. 3, pp. 1–18, 2017.

[27] V. M. Sreehari and D. K. Maiti, "Buckling and post buckling characteristics of laminated composite plates with damage under thermo-mechanical loading," Structures, vol. 6, 2016, doi: 10.1016/j.istruc.2016.01.002.

[28] M. Saadatmorad, R. A. Jafari-Talookolaei, M. H. Pashaei, and S. Khatir, "Damage detection on rectangular laminated composite plates using wavelet based convolutional neural network technique," Compos Struct, vol. 278, 2021, doi: 10.1016/j.compstruct.2021.114656.

[29] M. R. Ashory, A. Ghasemi-Ghalebahman, and M. J. Kokabi, "An efficient modal strain energy-based damage detection for laminated composite plates," Advanced Composite Materials, vol. 27, no. 2, 2018, doi: 10.1080/09243046.2017.1301069.

[30] L. W. Zhang, K. M. Liew, and J. N. Reddy, "Postbuckling analysis of bi-axially compressed laminated nanocomposite plates using the first-order shear deformation theory," Compos Struct, vol. 152, 2016, doi: 10.1016/j.compstruct.2016.05.040.

[31] B. Saiah, M. Bachene, M. Guemana, Y. Chiker, and B. Attaf, "On the free vibration behavior of nanocomposite laminated plates contained piece-wise functionally graded graphene-reinforced composite plies," Eng Struct, vol. 253, 2022, doi: 10.1016/j.engstruct.2021.113784.

[32] V. Zeighami and M. Jafari, "A closed-form solution for thermoelastic stress analysis of perforated asymmetric functionally graded nanocomposite plates," Theoretical and Applied Fracture Mechanics, vol. 118, 2022, doi: 10.1016/j.tafmec.2022.103251.

[33] M. Jafari and A. Rohani, "Optimization of perforated composite plates under tensile stress using genetic algorithm," J Compos Mater, vol. 50, no. 20, 2016, doi: 10.1177/0021998315612540.

[34] T. Farsadi, D. Asadi, and H. Kurtaran, "Fundamental frequency optimization of variable stiffness composite skew plates," Acta Mech, vol. 232, no. 2, 2021, doi: 10.1007/s00707-020-02871-9.

[35] L. W. Zhang and L. N. Xiao, "Mechanical behavior of laminated CNT-reinforced composite skew plates subjected to dynamic loading," Compos B Eng, vol. 122, 2017, doi: 10.1016/j.compositesb.2017.03.041.

[36] N. D. Duc, S. E. Kim, T. Q. Quan, D. T. Manh, and N. H. Cuong, "Nonlinear buckling of eccentrically stiffened nanocomposite cylindrical panels in thermal environments," Thin-Walled Structures, vol. 146, 2020, doi: 10.1016/j.tws.2019.106428.

[37] D. Balkan, Ö. Demir, and A. Arıkoğlu, "Dynamic analysis of a stiffened composite plate under blast load: A new model and experimental validation," Int J Impact Eng, vol. 143, 2020, doi: 10.1016/j.ijimpeng.2020.103591.

[38] D. Shi, D. He, Q. Wang, C. Ma, and H. Shu, "Free vibration analysis of closed moderately thick cross-ply composite laminated cylindrical shell with arbitrary boundary conditions," Materials, vol. 13, no. 4, 2020, doi: 10.3390/ma13040884.

[39] M. Zarei, G. H. Rahimi, and M. Hemmatnezhad, "Free vibrational characteristics of grid-stiffened truncated composite conical shells," Aerosp Sci Technol, vol. 99, 2020, doi: 10.1016/j.ast.2020.105717.

[40] R. Ansari and J. Torabi, "Numerical study on the buckling and vibration of functionally graded carbon nanotube-reinforced composite conical shells under axial loading," Compos B Eng, vol. 95, 2016, doi: 10.1016/j.compositesb.2016.03.080.

[41] L. W. Zhang, Z. G. Song, P. Qiao, and K. M. Liew, "Modeling of dynamic responses of CNT-reinforced composite cylindrical shells under impact loads," Comput Methods Appl Mech Eng, vol. 313, 2017, doi: 10.1016/j.cma.2016.10.020.

[42] E. Viola, L. Rossetti, N. Fantuzzi, and F. Tornabene, "Generalized stress–strain recovery formulation applied to functionally graded spherical shells and panels under static loading," Compos Struct, vol. 156, 2016, doi: 10.1016/j.compstruct.2015.12.060.

[43] J. Lu, Q. Yang, Z. Meng, K. Yang, W. Xu, and C. V. Chiu, "Modeling and dynamic analysis of functionally graded porous spherical shell based on Chebyshev-Ritz approach," Science and Engineering of Composite Materials, vol. 30, no. 1, 2023, doi: 10.1515/secm-2022-0214.

[44] G. Imbalzano, P. Tran, T. D. Ngo, and P. V. S. Lee, "A numerical study of auxetic composite panels under blast loadings," Compos Struct, vol. 135, 2016, doi: 10.1016/j.compstruct.2015.09.038.

[45] M. S. H. Al-Furjan et al., "A computational framework for propagated waves in a sandwich doubly curved nanocomposite panel," Eng Comput, vol. 38, no. 2, 2022, doi: 10.1007/s00366-020-01130-8.

[46] T. T. Nguyen, S. Yang, and J. Branke, "Evolutionary dynamic optimization: A survey of the state of the art," Swarm Evol Comput, vol. 6, 2012, doi: 10.1016/j.swevo.2012.05.001.

[47] G. Abdullah and Z. A. Haddi Hassan, "A Comparison between Genetic Algorithm and Practical Swarm to Investigate the Reliability Allocation of Complex Network," in Journal of Physics: Conference Series, 2021. doi: 10.1088/1742-6596/1818/1/012163.

[48] G. Guariso and M. Sangiorgio, "Improving the performance of multiobjective genetic algorithms: An elitism-based approach," Information (Switzerland), vol. 11, no. 12, 2020, doi: 10.3390/info11120587.

[49] C. Yang, Q. Qian, F. Wang, and M. Sun, "An improved adaptive genetic algorithm for function optimization," in 2016 IEEE International Conference on Information and Automation, IEEE ICIA 2016, 2017. doi: 10.1109/ICInfA.2016.7831905.

[50] S. Mahmoudinazlou and C. Kwon, "A hybrid genetic algorithm for the min–max Multiple Traveling Salesman Problem," Comput Oper Res, vol. 162, 2024, doi: 10.1016/j.cor.2023.106455.

[51] A. Kaveh, "Particle Swarm Optimization," in Advances in Metaheuristic Algorithms for Optimal Design of Structures, Cham: Springer International Publishing, 2017, pp. 11–43. doi: 10.1007/978-3-319-46173-1_2.

[52] D. Sedighizadeh, E. Masehian, M. Sedighizadeh, and H. Akbaripour, "GEPSO: A new generalized particle swarm optimization algorithm," Math Comput Simul, vol. 179, 2021, doi: 10.1016/j.matcom.2020.08.013.

[53] T. Li, J. Shi, W. Deng, and Z. Hu, "Pyramid particle swarm optimization with novel strategies of competition and cooperation," Appl Soft Comput, vol. 121, 2022, doi: 10.1016/j.asoc.2022.108731.

[54] D. Yazdani, D. Yazdani, D. Yazdani, M. N. Omidvar, A. H. Gandomi, and X. Yao, "A Species-based Particle Swarm Optimization with Adaptive Population Size and Deactivation of Species for Dynamic Optimization Problems," ACM Transactions on Evolutionary Learning and Optimization, vol. 3, no. 4, 2023, doi: 10.1145/3604812.

[55] S. Chakraborty, A. K. Saha, A. E. Ezugwu, J. O. Agushaka, R. A. Zitar, and L. Abualigah, "Differential Evolution and Its Applications in Image Processing Problems: A Comprehensive Review," Archives of Computational Methods in Engineering, vol. 30, no. 2, pp. 985–1040, Mar. 2023, doi: 10.1007/s11831-022-09825-5.

[56] W. L. Liu, Y. J. Gong, W. N. Chen, Z. Liu, and ..., "Coordinated charging scheduling of electric vehicles: a mixed-variable differential evolution approach," ... Transportation ..., 2019.

[57] L. Cui, G. Li, Q. Lin, J. Chen, and N. Lu, "Adaptive differential evolution algorithm with novel mutation strategies in multiple sub-populations," Comput Oper Res, vol. 67, 2016, doi: 10.1016/j.cor.2015.09.006.

[58] A. Draa, S. Bouzoubia, and I. Boukhalfa, "A sinusoidal differential evolution algorithm for numerical optimisation," Applied Soft Computing Journal, vol. 27, 2015, doi: 10.1016/j.asoc.2014.11.003.

[59] A. W. Mohamed and A. K. Mohamed, "Adaptive guided differential evolution algorithm with novel mutation for numerical optimization," International Journal of Machine Learning and Cybernetics, vol. 10, no. 2, pp. 253–277, Feb. 2019, doi: 10.1007/s13042-017-0711-7.

[60] Y. Li, M. Jia, X. Han, and X.-S. Bai, "Towards a comprehensive optimization of engine efficiency and emissions by coupling artificial neural network (ANN) with genetic algorithm (GA)," Energy, vol. 225, p. 120331, Jun. 2021, doi: 10.1016/j.energy.2021.120331.

[61] S. Sennan, S. Ramasubbareddy, S. Balasubramaniyam, A. Nayyar, M. Abouhawwash, and N. A. Hikal, "T2FL-PSO: Type-2 Fuzzy Logic-Based Particle Swarm Optimization Algorithm Used to Maximize the Lifetime of Internet of Things," IEEE Access, vol. 9, pp. 63966–63979, 2021, doi: 10.1109/ACCESS.2021.3069455.

[62] G. D'Angelo and F. Palmieri, "GGA: A modified genetic algorithm with gradient-based local search for solving constrained optimization problems," Inf Sci (N Y), vol. 547, pp. 136–162, Feb. 2021, doi: 10.1016/j.ins.2020.08.040.

[63] Ş. Karakaya and Ö. Soykasap, "Buckling optimization of laminated composite plates using genetic algorithm and generalized pattern search algorithm," Structural and Multidisciplinary Optimization, vol. 39, no. 5, 2009, doi: 10.1007/s00158-008-0344-2.

[64] L. Huang, C. T. Ng, A. H. Sheikh, and M. C. Griffith, "Niching particle swarm optimization techniques for multimodal buckling maximization of composite laminates," Applied Soft Computing Journal, vol. 57, 2017, doi: 10.1016/j.asoc.2017.04.006.

[65] A. R. Ghasemi, A. Tabatabaeian, M. H. Hajmohammad, and F. Tornabene, "Multi-step buckling optimization analysis of stiffened and unstiffened polymer matrix composite shells: A new experimentally validated method," Compos Struct, vol. 273, 2021, doi: 10.1016/j.compstruct.2021.114280.

[66] R. Kolahchi, S. P. Zhu, B. Keshtegar, and N. T. Trung, "Dynamic buckling optimization of laminated aircraft conical shells with hybrid nanocomposite martial," Aerosp Sci Technol, vol. 98, 2020, doi: 10.1016/j.ast.2019.105656.

[67] V. Ho-Huu, T. D. Do-Thi, H. Dang-Trung, T. Vo-Duy, and T. Nguyen-Thoi, "Optimization of laminated composite plates for maximizing buckling load using improved differential evolution and smoothed finite element method," Compos Struct, vol. 146, 2016, doi: 10.1016/j.compstruct.2016.03.016.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

[68] J. Liu, "Multi-objective optimization of multi-layered cylindrical shells with opening under axial load using the NSGA-II genetic algorithm," Journal of Intelligent and Fuzzy Systems, vol. 45, no. 6, 2023, doi: 10.3233/JIFS-230826.

[69] Khorramabadi Mahdi Karami and Nezamabadi Ali Reza, "Buckling Behavior of Functionally Graded Nano Clay-Reinforced Composite Beams," Journal of Aerospace Science and Technology, vol. 15, no. 2, pp. 110–117, 2022, doi: https://doi.org/10.22034/jast.2022.341613.1118.

[70] B. Keshtegar, T. Nguyen-Thoi, T. T. Truong, and S. P. Zhu, "Optimization of buckling load for laminated composite plates using adaptive Kriging-improved PSO: A novel hybrid intelligent method," Defence Technology, vol. 17, no. 1, 2021, doi: 10.1016/j.dt.2020.02.020.

[71] S. Moradi, A. R. Vosoughi, and N. Anjabin, "Maximum buckling load of stiffened laminated composite panel by an improved hybrid PSO-GA optimization technique," Thin-Walled Structures, vol. 160, 2021, doi: 10.1016/j.tws.2020.107382.

[72] X. Liu et al., "Design optimization of laminated composite structures using artificial neural network and genetic algorithm," Compos Struct, vol. 305, 2023, doi: 10.1016/j.compstruct.2022.116500.

[73] M. Savran and L. Aydin, "Natural frequency and buckling optimization considering weight saving for hybrid graphite/epoxy-sitka spruce and graphite-flax/epoxy laminated composite plates using stochastic methods," Mechanics of Advanced Materials and Structures, vol. 30, no. 13, 2023, doi: 10.1080/15376494.2022.2061656.

[74] A. R. Vosoughi, H. Dehghani Forkhorji, and H. Roohbakhsh, "Maximum fundamental frequency of thick laminated composite plates by a hybrid optimization method," Compos B Eng, vol. 86, 2016, doi: 10.1016/j.compositesb.2015.10.010.

[75] K. Kalita, R. K. Ghadai, and S. Chakraborty, "A comparative study on the metaheuristic-based optimization of skew composite laminates," Eng Comput, vol. 38, no. 4, 2022, doi: 10.1007/s00366-021-01401-y.

[76] S. Pal, K. Kalita, and S. Haldar, "Genetic Algorithm-Based Fundamental Frequency Optimization of Laminated Composite Shells Carrying Distributed Mass," Journal of The Institution of Engineers (India): Series C, vol. 103, no. 3, 2022, doi: 10.1007/s40032-021-00801-9.

[77] S. Peng, M. Habibi, and A. Pourjabari, "Generalized differential quadrature element solution, swarm, and GA optimization technique to obtain the optimum frequency of the laminated rotary nanostructure," Eng Anal Bound Elem, vol. 151, 2023, doi: 10.1016/j.enganabound.2023.02.052.

[78] L. Le-Anh, T. Nguyen-Thoi, V. Ho-Huu, H. Dang-Trung, and T. Bui-Xuan, "Static and frequency optimization of folded laminated composite plates using an adjusted Differential Evolution algorithm and a smoothed triangular plate element," Compos Struct, vol. 127, 2015, doi: 10.1016/j.compstruct.2015.02.069.

http://dx.doi.org/10.21622/ACE.2024.04.2.930

[79] T. Vo-Duy, T. Truong-Thi, V. Ho-Huu, and T. Nguyen-Thoi, "Frequency optimization of laminated functionally graded carbon nanotube reinforced composite quadrilateral plates using smoothed FEM and evolution algorithm," J Compos Mater, vol. 52, no. 14, 2018, doi: 10.1177/0021998317737831.

[80] N. D. Duc, K. Foroutan, S. M. Varedi-Koulaei, and H. Ahmadi, "Nonlinear Vibration Analysis of Laminated Composite Cylindrical Shell Under External Loading Utilizing Meta-Heuristic Optimization Algorithms," Iranian Journal of Science and Technology - Transactions of Mechanical Engineering, vol. 48, no. 2, 2024, doi: 10.1007/s40997-023-00685-3.

[81] R. Xiang, Z. Z. Pan, H. Ouyang, and L. W. Zhang, "A study of the vibration and lay-up optimization of rotating cross-ply laminated nanocomposite blades," Compos Struct, vol. 235, 2020, doi: 10.1016/j.compstruct.2019.111775.

[82] M. De Munck, S. De Sutter, S. Verbruggen, T. Tysmans, and R. F. Coelho, "Multi-objective weight and cost optimization of hybrid composite-concrete beams," Compos Struct, vol. 134, 2015, doi: 10.1016/j.compstruct.2015.08.089.

[83] M. Gholami, R. A. Alashti, and A. Fathi, "Optimal design of a honeycomb core composite sandwich panel using evolutionary optimization algorithms," Compos Struct, vol. 139, 2016, doi: 10.1016/j.compstruct.2015.12.019.

[84] S. Shrivastava, P. M. Mohite, T. Yadav, and A. Malagaudanavar, "Multi-objective multi-laminate design and optimization of a Carbon Fibre Composite wing torsion box using evolutionary algorithm," Compos Struct, vol. 185, 2018, doi: 10.1016/j.compstruct.2017.10.041.

[85] A. Albanesi, N. Roman, F. Bre, and V. Fachinotti, "A metamodel-based optimization approach to reduce the weight of composite laminated wind turbine blades," Compos Struct, vol. 194, 2018, doi: 10.1016/j.compstruct.2018.04.015.

http://dx.doi.org/10.21622/ACE.2024.04.2.951

LEVERAGING DEEP LEARNING TECHNOLOGY FOR ENHANCING PRINTING PRESS QUALITY

Omotunde Alabi Muyiwa[1], and B.H. Adejumo[2]

[1,2] Department of Industrial and Production Engineering, Faculty of Technology, University of Ibadan, Nigeria

Emails: {folatunde2004@gmail.com, adejumoboluwatife@gmail.com}

Received on, 06 August 2024 - Accepted on, 11 October 2024 - Published on, 22 October 2024

ABSTRACT

The use of machine learning techniques for printing quality control has not yet been widely adopted by most printing houses in Nigeria. Whereas deep learning technology can be used to improve printing quality. This study was designed to leverage deep learning technology for defect-detection in newspaper to improve printing quality. Six hundred newspaper images were loaded into a PyCharm programming environment for data exploration, cleaning, pre-processing, and augmentation. The MATPLOT library was used to analyse the visual characteristics of random samples from the image dataset. A four-hundred newspaper-images were selected, which were divided into 320 (160-defective+160 non-defective) for training, and 80 (40-defective+40 non-defective) for validation and testing. The Convolutional Neural Networks (CNN), Support Vector Machines (SVM), Gaussian filters, Local Binary Patterns (LBP), pre-trained Visual Geometry Group sixteen (VGG16) models, Neural Network Search (NNS), and Deep Forest Models (DFM) were used for defect-detections. The CNN achieved acceptable performance in image feature extraction for defect detection, with a validation accuracy of 66.7%. The machine learning ensemble classifiers of Gaussian filter+ LBP + SVM, CNN + SVM, simple CNN, transfer learning with VGG16, and NNS gave training accuracy of 97.3, 71.5, 72.5, 81.3, and 82.3%, respectively. These results demonstrate the effectiveness of various machine learning techniques for defect detection in newspaper images. The Gaussian filter+LBP+SVM model achieved highest accuracy, while its precision, recall and F1 score were 90.3, 89.4, and 85.9%, respectively. The printing press can leverage on deep learning models to improve quality of the newspaper printing.

Index words: Deep learning, Defect detection, Quality control, Newspaper industry.

I. INTRODUCTION

The increase in demand in printing industry for quality makes it necessary to research other techniques and methods that can replace the traditional way of defect detection that requires human visual inspection. Deep learning is a subset of machine learning and Artificial Intelligence (AI) that involves training artificial neural networks to learn from data, which is inspired by the structure and function of the human brain, specifically the way neurons communicate with each other. Deep learning has become a powerful tool for industries, enabling the analysis of large, complex datasets and facilitating decision-making processes that are often more accurate than those generated by humans. The growth of deep learning technology

http://dx.doi.org/10.21622/ACE.2024.04.2.951

is driven by the availability of big data, advances in neural network architectures, and powerful computing hardware.

Automation refers to the use of technology or computers to eliminate human involvement in tasks, thereby making work faster, easier, and reducing errors. Automation is a growing domain and computer programs such as deep learning algorithms are being developed to learn patterns from existing or historical data for better prediction and thereby enhance robust decision-making. This involves the use of systems, computer algorithms, or software to automate repetitive and monotonous tasks within organisations. Most printing press in Nigeria are yet to adopt deep learning technology to assess quality of printing for identification of defects or errors in printing processes.

Quality control is a mechanism by which printing press can attain the required specifications according to the regulatory body and customer expectations. It is concerned with making things factual rather than discovering and rejecting the printings. Deep learning technology has gained much applicability in almost every aspect of life due to its immense potential and capability to learn hierarchical features from various types of data, e.g., numerical, image, text, and audio, which made it a powerful tool in solving recognition problems (Wang et al., 2020), thereby, enabling continuous quality improvement. Deep learning is being used in numerous industries like health, pharmaceuticals, Information Technology (IT), and manufacturing. However, the use of deep learning technology in printing industry in Nigeria is not yet common. Hence, there is need to explore how leveraging on deep learning technology can enhance printing press quality in Nigeria.

II. MATERIAL

The use of X-ray Computed Tomography (XCT) to understand nondestructive evaluation of the Additively Manufactured (AM) parts impact on process parameters and quantifying the built part was found challenging. Hence, a deep learning network was trained using CAD models to experiment data obtained from XCT of an AM Jet engine turbine blade, which revealed promising preliminary results. Also, machine learning can be leveraged to obtain latest high-performance metamaterials and optmised topological designs. The delay in printing high-quality parts using bottom-up stereolithography was resolved using deep learning network (Khadilkar et al., 2019; Wang et al., 2020; Ziabari et al., 2021). Machine learning technology encouraged actionable intelligence through processing of collected data, thereby increasing manufacturing efficiency without noticeable difference in the required resources of humans or materials (Rai et al., 2021). The techniques of machine learning are being used to detect and prevent hacker threats in cybersecurity by leveraging on its data-driven styles to analyse large amount of information, detecting patterns and anomalies showing malicious activities (Shah, 2021). Lawson et al. (2021) stated that machine learning granted researchers a distinct chance in metabolic engineering for more predictability by leveraging omics data and improved production. Machine learning technology used an alogrithm that is able to learn autonomously through the direct data inputted (Bertolini et al.,2021). Zhang et al. (2019) applied Deep Convolutional Neural Network (DCNN) to classify printing defect into crystal, point, no printing, smears, overprinting, trailing, paper powder, and ink. The DCNN applied achieved 96.86% classification accuracy, compared to the deep transfer learning method, however, when combined the DCNN and SVM+SMOTE there was a 20% accuracy improvement. Dhanawat (2022) employed advanced algorithms and analytical knowledge of machine learning to discover intricate landscape of

http://dx.doi.org/10.21622/ACE.2024.04.2.951

anomaly detection within blockchain transactions, which performed accurately and efficiently in anomaly detection. Villalba-Diez et al. (2019) showed possibility of combined deep learning soft sensor application with high resolution optical quality control camera to increase quality accuracy and reduced amount of industrial visual inspection during printing process. Samepedro et al. (2021) observed that devices used in 3D printing do encounter undetected errors and problems, which can cause severe damage to the printer and resulted in output rejection. Therefore, the researchers evaluated different types of deep learning techniques like Multilayer Perception (MLP), Long Short Term Memory (LSTM), and Combine Convolutional Neural Networks (CNN) and LSTM, yet, application of the LSTM was found to be the best. In a combination of deep learning and brain computer interface for 3D where collected data were preprocessed in a MATLAB environment, thereafter used to train various neural network architectures, it was found that CNN-LSTM served the purpose of classfying objects accurately (Kachhia et al., 2020). The present study is using pyCharm programmed environment for data exploration, cleaning, pre-processing, augmentation, while MATPLOT library was used to analyse visual characteristics of loaded random sample image-dataset. Yao et al. (2024) considered conductivity and physical defects in refined print line quality, therefore, a model for defect detection was built using neural architecture and within 4.6ms the model detected an image with an accuracy of 95.5%. Tsai et al. (2019) identified unlawful tappering of printed documents, fake currency, and copyright violation to develop an efficient and safe testing instrument to know source of printed materials using CNN of deep learning capable of learning features automatically and found that feature based support vector machine performed better than deep learning system. The CNN fault diagnosis adopted in printer using 3D to detect and categorise glitches in printing revealed significant precision using secondary data, while support vector machine (5.1%) and artificial neural network (25.7%) provided less precision results compared with the CNN (Verana et al., 2021). Im et al. (2022) suggested that bias can occur in analysing images because classification of images require expertise, therefore, developed a deep learning classification systems, which was found to achieve good results. Hence, in this present study deep learning technology is being leveraged for quality printing in newspaper industry in Nigeria.

III. MEHODS

Through data exploration and visualisation of 600 sample images of printed newspaper, the image data were loaded into pyCharm programming environment. Subsequently, a random sample of the images from the dataset was displayed using a grid from the Matplotslib library to analyse the visual characteristics. The printing paper defects found were smudging, blurring, fading, and colour misregistration. Data cleaning and pre-processing were done by resizing collected images to a standardised size suitable for model input. The input size was set to 256 by 256 pixels to achieve a balance between captured details and minimised scaling operations, in order to convert categorical labels into numerical format, and normalised pixel values to a common scale (e.g., 0 to 1) for consistent model training. Data augmentation was done to prevent over-fitting and enhance generalisation ability of the machine learning models. Transformations such as rotation, flipping, and zooming were applied to ensure variability and diversity in the training data, helping the model better adapt to different patterns. After data cleaning, pre-processing and augmentation of the 400 pages were prepared as dataset for analysis. The dataset was divided into 320 training images (160 defective + 160 non-defective), and 80 (40 defective + 40 non-defective) for validation and testing sets. Training sets were utilised for estimating various parameters and evaluating the performance of the model. The validation set was employed during training process to fine-tune model parameters and prevent over-fitting. Upon splitting the data into these sets, a balanced distribution of "defective" and "non-defective" newspapers in each set was ensured. This balance helped to prevent bias and ensured that the model was a representative sample of both defective and non-defective instances during training, validation, and testing. The machine learning techniques employed for defect detection were the Support Vector Machine (SVM), Deep Forest Models (DFM), Convolutional Neural Network (CNN), and ensemble classifiers. The newspaper feature extractions were performed using Gaussian filter, Local Binary Pattern (LBP) and Laplacian of Gaussian filer to denoise images, and extract edges for input into the support vector machine. The Visual Geometry Group sixteen (VGG16), a pre-trained deep learning model, was performed on the ImageNet dataset. The DFM, which comprised cascade and multi-grained forest structures, were used to train the ensemble classifier. These were chosen based on their effectiveness and suitability for this present study. The models leverage ensemble learning techniques to combine predictions of multiple individual classifiers to enhance printing quality in the newspaper industry through deep learning technology. Fig 1. shows the proposed method procedure.

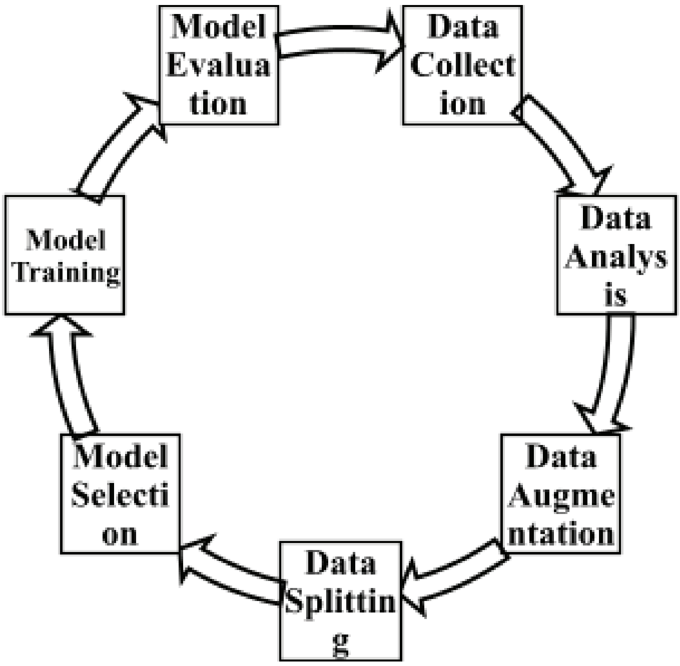


Fig. 1. Proposed method procedure

A. DATA EXPLORATION

This is summarising, visualising, and understanding the main features of the dataset. Its goal was to gain insights into the patterns, distributions, and relationships within the data before proceeding to more advanced analyses or modelling. The visualisation of sample images was done by loading the image data into pyCharm programming environment and subsequently displaying random sample of images from the dataset using a grid from the MATPLOT Library to analyse the visual characteristics.

B. DATA CLEANING AND PRE-PROCESSING

This was done to prepare the dataset for analysis by resizing collected images to

http://dx.doi.org/10.21622/ACE.2024.04.2.951

standardised size suitable for model input. The input size was set to 256 × 256 pixels to strike a balance between captured details and minimised scaling operations, in order to convert categorical labels into numerical format, and then normalised pixel values to a common scale (e.g., 0 to 1) for consistent model training.

C. DATA AUGMENTATION

This was done to prevent overfitting and enhance the generalisation ability of the machine learning model. Various transformations such as rotation, flipping, and zooming were introduced to ensure variability and diversity to the training data, thereby the model was better adapted to different patterns. This was performed at real time on the Central Processing Unit (CPU), which handles the generation of augmented samples, while the core training operations occurred on the Graphics Processing Unit (GPU).

D. DATA SPLITTING

The obtained dataset was divided into training, testing, and validation sets, which is a common practice in machine learning techniques. Training sets were utilised for estimating various parameters and evaluating the performance of the model. It serves as basis for the model to learn patterns and relationships within the data. Data validation set was employed during the training process to fine-tune model parameters and prevent overfitting. It helped to gauge the model generalisation ability to unseen data, hence, the deep learning model capacity to perform well on new and previously unseen data that were not explicitly trained on. After the training completion, the testing dataset was employed to assess the model performance on completely new and independent data. This step was intended to assess how well the model generalizes to real-world scenarios. By splitting the data into sets, a balanced distribution of "non-defective" and "defective" newspapers in each set was ensured. This balance helped prevent biases and ensured that the model was exposed to a representative sample of both positive and negative instances during training, testing and validation.

E. MODEL SELECTION

The rationale for choosing Convolutional Neural Networks (CNNs) lied in its well-suited architecture for classification tasks, particularly when dealing with structured grid data like images. The CNNs are designed with layers that perform convolutional operations, allowing it to automatically learn hierarchical features from input images. The CNNs has been found to be effective in image classification. Its ability to capture spatial hierarchies and local patterns through convolutional layers made it particularly powerful for tasks where understanding the visual context is crucial. The layered structure of the CNNs enabled it to automatically learn and extract features from images, making it highly effective in discerning patterns and making accurate classifications.

F. MODEL TRAINING

The training phase involved the datasets being introduced into the model that has been labelled or annotated, allowing the model to learn the intricate patterns and relationships present in the data. This learning process involves adjusting the model internal parameters or hyperparameters. In order to optimise the model performance and ensure its ability to generalise well to unseen data, a separate validation set was

http://dx.doi.org/10.21622/ACE.2024.04.2.951

used. The model parameters was fine-tuned based on its performance on validation set. The goal is to minimise the difference or error between the predictions made by the model and the actual target values in the validation set. This iterative process of learning from the training set and refining based on the validation set continues until the model achieves a satisfactory level of performance.

G. MODEL EVALUATION

The trained model was rigorously evaluated using various metrics to gain insights into its overall performance, strengths, and weaknesses. This evaluation was conducted on a separate test set to assess the model ability to generalise. Metrics such as accuracy, precision, recall, and F1 score were employed to provide comprehensive understanding of the model capabilities. In order to assess the model defect detection capabilities, the dataset went through stratified random sampling, creating distinct training, validation and testing sets. The training set was utilised for model training, while validation set determine optimal training epochs and decision thresholds. The test set evaluated the model out-of-sample performance for defect detection.

1. HARDWARE AND SOFTWARE REQUIREMENTS

Computer Processing Unit (CPU): Intel Xeon E5-2680v4 running at 2.40GHz with 14 cores.

Graphic Processing Unit (GPU): NVIDIA Tesla P100 with 16GB of memory, providing significant acceleration for machine learning models.

Random Access Memory (RAM): 128GB, ensuring efficient handling of large datasets and complex models without encountering memory issues.

Operating System: Ubuntu 16.04 LTS, chosen for its stability, compatibility with a wide range of machine learning libraries, and ease of use.

Python Interpreter: Python 3.6.5, the standard language for machine learning and data science, ensuring compatibility with the required libraries.

Gradient boosted Collaborative Forest (gcForest) Library: Version 1.0.9 of the gcForest library, suitable for defect detection tasks and also known for its effectiveness in implementing deep forest models.

Other Libraries: Numpy 1.16.4, Scipy 1.2.1, Scikit-learn 0.20.3, fundamental for data manipulation, scientific computing and machine learning tasks.

IV. RESULTS AND DISCUSSION

Figure 2. Shows the classification of the 320 newspapers images (160-defective + 160 non-defective) and 80 newspaper images (40-defective and 40 non-defective) to belong to two classes.

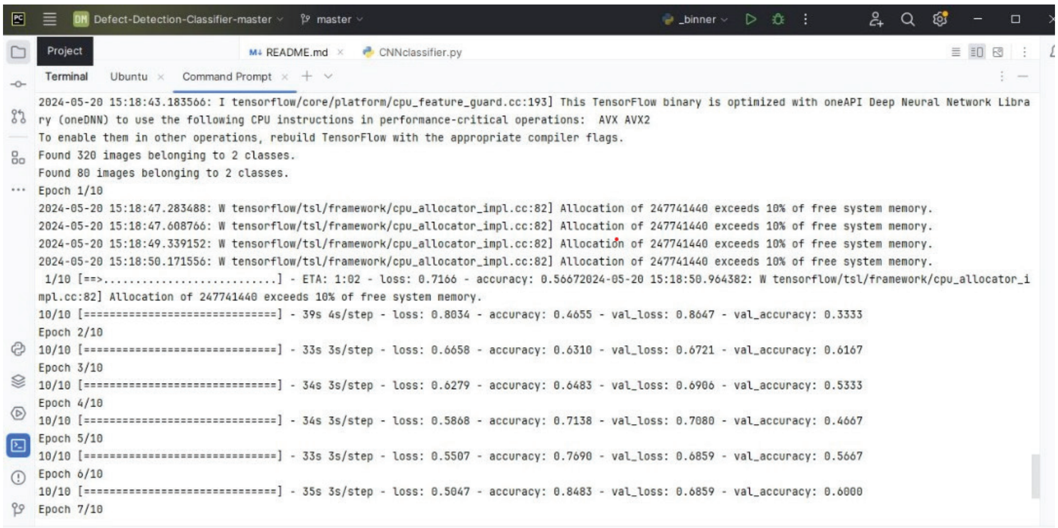


Fig. 2. Defect-Detection classifier

Figure 3. presents results of defect-detection classifiers training loss and accuracy, and validation loss and accuracy. Figure 3. displayed step by step ten epoch for training loss, accuracy, validation loss and accuracy for the 400 newspaper images selected and characterised.

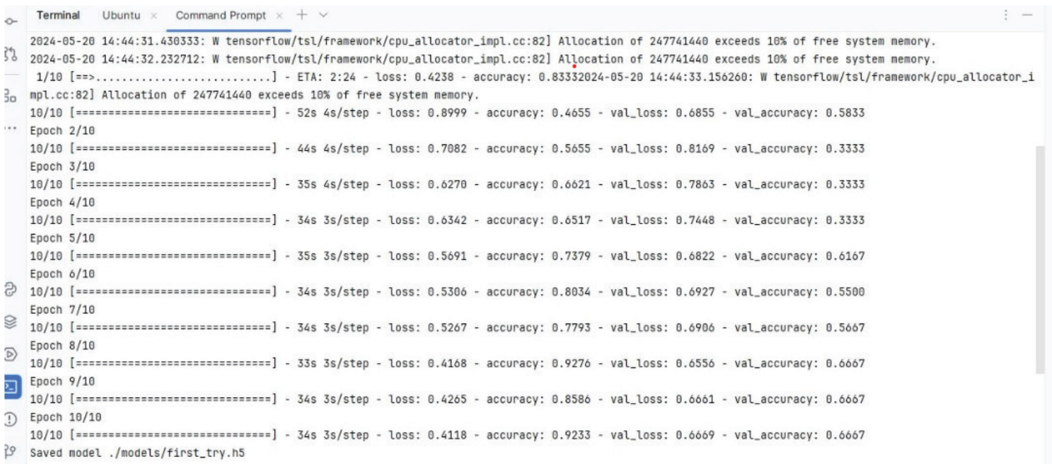


Fig. 3. Defect-Detection classifier result

Table I presents Convolutional Neural Network (CNN) architecture over 10 epochs of trained and validated data for deep learning model and achieved a training accuracy of 88% and a validation loss and accuracy balanced at 67%. The loss values for both trained and validated datasets decreased gradually over the 10 epochs, which indicated that the model learnt the features of the dataset. The validation accuracy started from 32% and improved to 67%. The training loss decreases from 1.14 to 0.46 over the 10 epochs, which was a good indication that the model was learning. This represents a reasonable improvement based on the data used in the present study.

http://dx.doi.org/10.21622/ACE.2024.04.2.951

TABLE I
CONVOLUTION NEURAL NETWORK MODEL TRAINING AND VALIDATION ACCURACY RESULTS

Epoch	Training loss	Training accuracy	Validation loss	Validation accuracy
1/10	1.14	0.47	0.81	0.32
2/10	0.86	0.53	0.67	0.73
3/10	0.68	0.68	0.72	0.33
4/10	0.65	0.68	0.72	0.35
5/10	0.61	0.73	0.71	0.38
6/10	0.55	0.82	0.70	0.48
7/10	0.58	0.81	0.69	0.60
8/10	0.53	0.80	0.67	0.65
9/10	0.51	0.84	0.66	0.67
10/10	0.46	0.88	0.67	0.67

Table II presents Convolutional Neural Network (CNN) architecture over 10 epochs of trained and validated data for deep learning model and achieved a training recall of 94%. The validation loss, and recall of 64 and 0%, respectively. The loss values for both trained and validated datasets decreased gradually over the 10 epochs, which indicated that the model learnt the features of the dataset. The validation recall started at 1.00 and dropped to 0.00, indicating no improvement over the 9 epochs. However, training recall increases from 0.34 to 0.94 over the 10 epochs, which indicated that the model was learning.

TABLE II
CONVOLUTIONAL NEURAL NETWORK MODEL TRAINING AND VALIDATION RECALL RESULTS

Epoch	Training loss	Training recall	Validation loss	Validation recall
1/10	1.07	0.34	0.94	1.00
2/10	0.74	0.43	0.66	0.00
3/10	0.65	0.54	0.65	0.00
4/10	0.61	0.67	0.63	0.00
5/10	0.61	0.64	0.63	0.00
6/10	0.54	0.82	0.64	0.00
7/10	0.52	0.81	0.64	0.00
8/10	0.48	0.85	0.64	0.00
9/10	0.45	0.92	0.63	0.00
10/10	0.44	0.94	0.64	0.00

Table III presents Convolutional Neural Network (CNN) architecture over 10 epochs of trained and validated data for deep learning model and achieved a training precision of 71%, and validation loss, and precision of 73 and 33%, respectively. The loss values for both trained and validated datasets decreased gradually over the 10 epochs, which indicated that the model learnt the features of the dataset. The

http://dx.doi.org/10.21622/ACE.2024.04.2.951

validation precision started from 0.00% and rises to a steady value of 33% over the 10 epochs. However, training precision increases from 0.49 to 0.71 over the 10 epochs, which indicated that the model was learning.

TABLE III
CONVOLUTIONAL NEURAL NETWORK MODEL TRAINING AND VALIDATION PRECISION RESULTS

Epoch	Training loss	Training precision	Validation loss	Validation precision
1/10	1.39	0.49	0.57	0.00
2/10	0.90	0.49	0.89	0.33
3/10	0.75	0.47	0.94	0.33
4/10	0.69	0.59	0.89	0.33
5/10	0.70	0.58	0.86	0.33
6/10	0.65	0.63	0.84	0.33
7/10	0.63	0.65	0.82	0.33
8/10	0.62	0.64	0.79	0.33
9/10	0.60	0.67	0.75	0.33
10/10	0.58	0.71	0.73	0.33

Table IV presents Convolutional Neural Network (CNN) architecture over 10 epochs of trained and validated data for deep learning model and achieved a training F1 score of 91%. The validation loss, and F1 score of 69% and 23%, respectively. The loss values for both trained and validated datasets decreased gradually over the 10 epochs, which indicated that the model learnt the features of the dataset. The validation F1 score started from 0.50 steady till the 3rd epoch and dropped to 0.19 at 4th epoch, 0.00 from 5 to 8th epoch, at 9th epoch it was 0.58 and dropped to 0.23 at the 10th epoch, indicating that the model effectiveness over the 10 epochs is not steady. However, training F1 score increases from 0.63 to 0.91 over the 10 epochs, which indicated that the model was learning.

TABLE IV
CONVOLUTIONAL NEURAL NETWORK MODEL TRAINING AND VALIDATION F1 SCORE RESULTS

Epoch	Training loss	Training F1 Score	Validation loss	Validation F1 Score
1/10	0.79	0.63	0.76	0.50
2/10	0.69	0.66	0.76	0.50
3/10	0.63	0.76	0.74	0.50
4/10	0.62	0.71	0.71	0.19
5/10	0.58	0.77	0.69	0.00
6/10	0.52	0.87	0.68	0.00
7/10	0.49	0.85	0.67	0.00
8/10	0.47	0.85	0.67	0.00
9/10	0.44	0.91	0.69	0.58
10/10	0.42	0.91	0.69	0.23

http://dx.doi.org/10.21622/ACE.2024.04.2.951

Table V presents Gauss filter LBPSVM architecture over 10 epochs of trained and validated data for deep learning model and achieved a training recall of 89%. The validation loss and recall of 91 and 80 %, respectively at 9th epoch. The model training loss decreased gradually over the 9 epochs, while validation loss fluctuated, which indicated that the model learnt the features of the dataset. The validation recall started from 1.00 and decreased gradually from the 5th epoch after few instabilities in the upper epoch, indicating the model detected fewer defects in the printing, while training recall decreases from 0.76 to 0.49 for 1-4 epoch, but increased from 0.64 to 0.89 for 6-10 epochs.

TABLE V
GAUSSFILTER+LBP+SVM MODEL TRAINING AND VALIDATION RECALL RESULTS

Epoch	Training loss	Training recall	Validation loss	Validation recall
1/10	1.52	0.76	0.70	1.00
2/10	0.69	0.82	0.69	0.00
3/10	0.68	0.44	0.69	0.03
4/10	0.69	0.49	0.69	1.00
5/10	0.62	0.68	0.63	0.92
6/10	0.63	0.64	0.68	0.98
7/10	0.54	0.81	0.67	0.83
8/10	0.38	0.82	0.73	0.83
9/10	0.35	0.84	0.91	0.80
10/10	0.29	0.89	0.82	0.63

Table VI presents GaussfilterLBPSVM architecture over 10 epochs of trained and validated data for deep learning model and achieved a training precision of 90%, and average validation loss, and precision of 73.50% and 66.60%, respectively.

TABLE VI
GAUSS FILTER + LBP + SVM MODEL TRAINING AND VALIDATION PRECISION RESULTS

Epoch	Training loss	Training precision	Validation loss	Validation precision
1/10	2.18	0.42	0.70	0.00
2/10	0.70	0.58	0.69	0.58
3/10	0.70	0.61	0.69	0.83
4/10	0.67	0.63	0.68	0.64
5/10	0.62	0.68	0.62	0.69
6/10	0.61	0.69	0.69	0.78
7/10	0.48	0.73	0.95	0.72
8/10	0.43	0.82	0.77	0.77
9/10	0.35	0.81	0.76	0.74
10/10	0.41	0.90	0.80	0.61

Table VII presents Gauss filter LBPSVM architecture over 10 epochs for F1 Score. The F1 score starts at 0.38 in the first epoch and improves significantly, reaching 0.86

by the 10th epoch. This indicates that the model is getting better at handling both precision and recall for the training data, leading to an overall improved performance in detection. The loss consistently decreases, starting at 1.21 in the first epoch and reaching 0.30 by the last epoch. A decreasing loss is a positive sign, indicating that the model is learning and fitting the training data better over time. The F1 score on the validation set starts at 0.00 in the first epoch, suggesting that the model was initially unable to generalize well on unseen data. By the second epoch, the validation F1 score jumps to 0.43 but fluctuates in the subsequent epochs. After reaching a peak around 0.46 in the fourth epoch, the F1 score starts declining until 0.29 by the 10th epoch. The drop in validation F1 score suggests that the model is likely overfitting but performs well on the training data, while model struggles to generalize to the validation set. The validation loss initially decreases slightly, reaching a low point in the third epoch but starts to increase steadily from the 4th epoch at 0.68 to 1.16 in the final epoch. The increase in validation loss, along with the drop in the F1 score indicates overfitting, however, the Gauss filter + LBP + SVM model is learning patterns well from the training data. Nevertheless, in order for the model to generalise to the unseen data there is needs for more data for the training of the model to be applied in a large-scale printing industry apart from this present study on one stop shop printing press.

TABLE VII
GAUSS FILTER + LBP + SVM MODEL TRAINING AND VALIDATION F1 SCORE RESULTS

Epoch	Training loss	Training F1 Score	Validation loss	Validation F1 Score
1/10	1.21	0.38	0.69	0.00
2/10	0.69	0.50	0.68	0.43
3/10	0.67	0.65	0.67	0.36
4/10	0.62	0.68	0.68	0.46
5/10	0.55	0.68	0.70	0.43
6/10	0.51	0.73	0.71	0.41
7/10	0.47	0.80	0.74	0.42
8/10	0.36	0.87	0.88	0.28
9/10	0.31	0.88	0.94	0.30
10/10	0.30	0.86	1.16	0.29

Table VIII presents ensemble classifiers and their training accuracy for newspaper images extracted for evaluation to detect any defection using Gaussian filter plus Local Binary Pattern (LBP), plus Support Vector Machine (SVM) with a radial basis function (rbf) {Gauss filter + LBP + SVM (rbf)}, which achieved an accuracy of 97.3%. The Convolution Neural Network (CNN) + SVM (rbf kernel) achieved an accuracy of 71.3%. Also, simple CNN (3Convolution + 1FC) achieved an accuracy of 72.5%, and transfer learning (VGG16) achieved an accuracy of 81.3%, Neural Network Search (NNS) achieved an accuracy of 82.3%. The NNS technique automatically searched for optimal neural network architecture for a given task compared to manually designed architectures. These results demonstrated effectiveness of different machine learning techniques for defect detection in newspaper images. However, Gauss filter+LBP+SVM (rbf kernel) revealed highest accuracy (97.3%), which means its learning rate was very high compared with other designed architectures.

http://dx.doi.org/10.21622/ACE.2024.04.2.951

TABLE VIII
COMPARISON OF ENSEMBLE CLASSIFIERS AND THEIR RECEPTIVE TRAINING ACCURACIES

Classifier	Accuracy
Gauss filter+LBP+SVM (rbf kernel)	97.3%
CNN+SVM (rbf kernel)	71.3%
Simple CNN (3 Conv+1 FC)	72.5%
Transfer Learning (VGG16)	81.3%
Neural Network Search	82.3%

Table IX presents ensemble classifiers and their training precision for newspaper images extracted for evaluation to detect any defection using Gaussian filter plus Local Binary Pattern (LBP), plus Support Vector Machine (SVM) with a radial basis function (rbf) {Gauss filter + LBP + SVM (rbf)}, which achieved precision of 90.3%. The Convolution Neural Network (CNN) + SVM (rbf kernel) achieved precision of 84.0%. Also, simple CNN (3Convolution + 1FC) achieved precision of 71.2%, and transfer learning (VGG16) achieved precision of 88.4%, Neural Network Search (NNS) achieved precision of 90.7%. The NNS technique automatically searched for optimal neural network architecture for a given task compared to manually designed architectures. These results demonstrated precision of different machine learning techniques for defect detection in newspaper images. However, the auto NNS shows precision of 90.7% compared to Gauss filter+LBP+SVM (rbf kernel) of 90.3%. Therefore, the learning technique of auto NNS shows fractional difference to Gauss filter+LBP+SVM (rbf kernel).

TABLE IX
COMPARISON OF ENSEMBLE CLASSIFIERS AND THEIR RESPECTIVE TRAINING PRECISIONS

Classifier	Precision
Gauss filter+LBP+SVM (rbf kernel)	90.3%
CNN+SVM (rbf kernel)	84.0%
Simple CNN (3 Conv+1 FC)	71.2%
Transfer Learning (VGG16)	88.4%
Neural Network Search	90.7%

Table X presents ensemble classifiers and their training recall for newspaper images extracted for evaluation to detect any defection using Gaussian filter plus Local Binary Pattern (LBP), plus Support Vector Machine (SVM) with a radial basis function (rbf) {Gauss filter + LBP + SVM (rbf)}, gave recall of 89.4%. The Convolution Neural Network (CNN) + SVM (rbf kernel) gave recall of 95.6%. Also, simple CNN (3Convolution + 1FC) achieved recall of 93.9%, and transfer learning (VGG16) achieved precision of 90.0%, Neural Network Search (NNS) achieved precision of 97.5%. The NNS technique automatically searched for optimal neural network architecture for a given task compared to manually designed architectures. These results demonstrated recall of different machine learning techniques for defect detection in newspaper images. However, the auto NNS, CNN+SVM, simple CNN, transfer learning, and Gauss filter+LBP+SVM (rbf kernel) shows recall of 97.5, 95.6, 93.9, 90.0, and 89.4%, respectively.

TABLE X
COMPARISON OF ENSEMBLE CLASSIFIERS AND THEIR RECEPTIVE TRAINING CLASSIFIER

Classifier	Recall
Gauss filter+LBP+SVM (rbf kernel)	89.4%
CNN+SVM (rbf kernel)	95.6%
Simple CNN (3 Conv+1 FC)	93.9%
Transfer Learning (VGG16)	90.0%
Neural Network Search	97.5%

Table XI presents ensemble classifiers and their training F1 score for newspaper images extracted for evaluation to detect any defection using Gaussian filter plus Local Binary Pattern (LBP), plus Support Vector Machine (SVM) with a radial basis function (rbf) {Gauss filter + LBP + SVM (rbf)}, Convolution Neural Network (CNN) + SVM (rbf kernel), simple CNN (3Convolution + 1FC), transfer learning (VGG16), and Neural Network Search (NNS) reached F1score of 85.9, 84.0, 91.5, 93.7, and 90.5%, respectively. The NNS technique automatically searched for optimal neural network architecture for a given task compared to manually designed architectures. The transfer learning (VGG16) indicates highest recall (93.7%) compared to other machine learning techniques.

TABLE XI
COMPARISON OF ENSEMBLE CLASSIFIERS AND THEIR RESPECIVE TRAINING F1 SCORE

Classifier	F1 Score
Gauss filter+LBP+SVM (rbf kernel)	85.9%
CNN+SVM (rbf kernel)	84.0%
Simple CNN (3 Conv+1 FC)	91.5%
Transfer Learning (VGG16)	93.7%
Neural Network Search	90.5%

The hyper-parameter search and values for trial runs for the deep learning model were presented in Fig. 4. The image processing pipeline includes a vanilla block for standard convolutional layers. The inputted images were normalized, but not augmented. The convolutional layer in the block used a kernel size of 3, with 1 block and 2 layers per block. Max pooling was applied, but separable convolutions were not used. A dropout rate of 0.25 was applied, while the layer consists of 32 and 64 filters in the first and second layers, respectively. The classification head used a spatial reduction method of flatten, and applied a dropout rate of 0.5. The model was trained using adam optimizer with a learning rate of 0.001. The results from the trial indicated that the chosen hyper-parameters led to promising performance in classifying defects in newspapers.

http://dx.doi.org/10.21622/ACE.2024.04.2.951


```
Search: Running Trial #1

Value           |Best Value So Far |Hyperparameter
vanilla         |vanilla           |image_block_1/block_type
True            |True              |image_block_1/normalize
False           |False             |image_block_1/augment
3               |3                 |image_block_1/conv_block_1/kernel_size
1               |1                 |image_block_1/conv_block_1/num_blocks
2               |2                 |image_block_1/conv_block_1/num_layers
True            |True              |image_block_1/conv_block_1/max_pooling
False           |False             |image_block_1/conv_block_1/separable
0.25            |0.25              |image_block_1/conv_block_1/dropout
32              |32                |image_block_1/conv_block_1/filters_0_0
64              |64                |image_block_1/conv_block_1/filters_0_1
flatten         |flatten           |classification_head_1/spatial_reduction_1/reduction_type
0.5             |0.5               |classification_head_1/dropout
adam            |adam              |optimizer
0.001           |0.001             |learning_rate
```

Fig. 4. Hyper-parameter search

A. LIMITATIONS

This present model training was based on small scale data for one stop shop printing press which can be deployed at acquiring small size data for printing quality enhancement, however, for generalisation to large scale printing press for unseen data, more data are needed to train the model to improve its predictions.

V. CONCLUSION

The convolutional neural network deep learning models were shown to be effective in learning and detecting defects in newspaper images. The validation accuracies indicate that the models can generalize reasonably well to unseen data, which is crucial for real-world applications. The combined deep learning ensemble of Gaussian filter plus Local Binary Pattern (LBP) plus Support Vector Machine (SVM) with a radial basis function (rbf) {Gauss filter + LBP + SVM (rbf)}, Convolution Neural Network (CNN) + SVM (rbf kernel), simple CNN (3Convolution + 1FC), transfer learning (VGG16), Neural Network Search (NNS, which is a technique that automatically searched for optimal neural network architecture for a given task compared to manually designed architectures). The results revealed that Gauss filter+LBP+SVM (rbf kernel) was most effective among the different deep learning techniques used for defect detection of the newspaper images. This means that its learning rate was very high compared with other designed architectures of the deep learning technology. The NNS and Gauss filter + CNN + SVM gave highest precision indicating that both avoided false defect detection compared to other models. The auto NNS, and CNN + SVM gave highest recall values meaning that both models could identify nearly all defects with fewer undetected defects in the newspaper images. The F1 score revealed that transfer learning is the best performing model followed by simple CNN, while the Gauss filter + CNN + SVM F1 score indicates that the model did not balance precision and recall very well when it is compared to other models. In a hyper-parameter tuning performed using AutoKeras to optimise the model performance, the best hyper-parameters found during the search process involved using a vanilla block type, normalising the images but not augmenting the data, and using a specific setting for the convolutional layers and optimiser. The hyper-parameter tuning results indicated that the chosen parameters were effective for the newspaper defect detection task. The Gaussian filter+ LBP + SVM deep learning model can be adopted in printing press industry for high quality printing, based on its accuracy. This revealed that printing press can leverage on deep learning technology for enhancing printing quality in the printing press industry.

VI. REFERENCES

[1] M. Bertolini, D. Mezzogori, M. Neroni, and F. Zammori, "Machine Learning for industrial applications: A comprehensive literature review," 2021. doi: 10.1016/j.eswa.2021.114820.

[2] V. Dhanawat, "Anomaly detection in financial transactions using machine learning and blockchain technology," *International Journal of Business Management and Visuals*, vol. 5, no. 1, pp. 34–41, 2022.

[3] C. Im, Y. Kim, and T. Mandl, "Deep learning for historical books: classification of printing technology for digitized images," *Multimed Tools Appl*, vol. 81, no. 4, 2022, doi: 10.1007/s11042-021-11754-7.

[4] J. Kachhia, R. Natharani, and K. George, "Deep Learning Enhanced BCI Technology for 3D Printing," *in 2020 11th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2020*, 2020. doi: 10.1109/UEMCON51285.2020.9298124.

[5] A. Khadilkar, J. Wang, and R. Rai, "Deep learning–based stress prediction for bottom-up SLA 3D printing process," *International Journal of Advanced Manufacturing Technology*, vol. 102, no. 5–8, 2019, doi: 10.1007/s00170-019-03363-4.

[6] C. E. Lawson *et al.*, "Machine learning for metabolic engineering: A review," 2021. doi: 10.1016/j.ymben.2020.10.005.

[7] R. Rai, M. K. Tiwari, D. Ivanov, and A. Dolgui, "Machine learning in manufacturing and industry 4.0 applications," 2021. doi: 10.1080/00207543.2021.1956675.

[8] G. A. Sampedro, M. A. Paramartha Putra, D. S. Kim, and J. M. Lee, "3D Printer State Prediction: A Deep Learning Model Approach," in *Proceedings - 2021 1st International Conference in Information and Computing Research*, iCORE 2021, 2021. doi: 10.1109/iCORE54267.2021.00043.

[9] V. Shah, "Machine Learning Algorithms for Cybersecurity: Detecting and Preventing Threats," *Revista Espanola de Documentacion Cientifica*, vol. 15, no. 4, pp. 42–66, 2021.

[10] M.-J. Tsai, Y.-H. Tao, and I. Yuadi, "Deep learning for printed document source identification," *Signal Process Image Commun*, vol. 70, pp. 184–198, Feb. 2019, doi: 10.1016/j.image.2018.09.006.

[11] M. Verana, C. I. Nwakanma, J. M. Lee, and D. S. Kim, "Deep Learning-Based 3D Printer Fault Detection," in *2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN)*, IEEE, Aug. 2021, pp. 99–102. doi: 10.1109/ICUFN49451.2021.9528692.

http://dx.doi.org/10.21622/ACE.2024.04.2.951

[12] J. Villalba-Diez, D. Schmidt, R. Gevers, J. Ordieres-Meré, M. Buchwitz, and W. Wellbrock, "Deep Learning for Industrial Computer Vision Quality Control in the Printing Industry 4.0," *Sensors*, vol. 19, no. 18, p. 3987, Sep. 2019, doi: 10.3390/s19183987.

[13] C. Wang, X. P. Tan, S. B. Tor, and C. S. Lim, "Machine learning in additive manufacturing: State-of-the-art and perspectives," *Addit Manuf*, vol. 36, p. 101538, Dec. 2020, doi: 10.1016/j.addma.2020.101538.

[14] C. Yao, L. Wang, Q. Wang, Z. Liu, G. Liu, and M. Zhang, "Deep-Learning-Guided Evaluation Method for the High-Volume Preparation of Flexible Sensors Based on Inkjet Printing," *ACS Appl Mater Interfaces*, vol. 16, no. 10, pp. 13326–13334, Mar. 2024, doi: 10.1021/acsami.4c00322.

[15] A. Ziabari *et al.*, "Beam Hardening Artifact Reduction in X-Ray CT Reconstruction of 3D Printed Metal Parts Leveraging Deep Learning and CAD Models," in *Volume 2B: Advanced Manufacturing*, American Society of Mechanical Engineers, Nov. 2020. doi: 10.1115/IMECE2020-23766.

[16] E. Zhang, B. Li, P. Li, and Y. Chen, "A Deep Learning Based Printing Defect Classification Method with Imbalanced Samples," *Symmetry (Basel)*, vol. 11, no. 12, p. 1440, Nov. 2019, doi: 10.3390/sym11121440.

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

DEVELOPMENT OF POLICY RESEARCH-EVIDENCE ORGANIZER AND PUBLIC HEALTH-POLICY EVALUATION TOOL (PROPHET): A COMPUTING PARADIGM FOR PROMOTING EVIDENCE-INFORMED POLICYMAKING IN NIGERIA

Kingsley Otubo Igboji[1,2,*], Chigozie Jesse Uneke[2], Fergus U. Onu[3], and Onyedikachi Chukwu[2]

[1] Department of Computer Science, David Umahi Federal University of Health Sciences, Uburu, Nigeria
[2] African Institute for Health Policy & Health Systems, Ebonyi State University, Abakaliki, Nigeria
[3] Department of Computer Science, Ebonyi State University, Abakaliki, Nigeria

Emails: {otubok@yahoo.com, unekecj@yahoo.com, uchefergus@yahoo.com, drchuksonyedika@yahoo.com}

Received on, 30 October 2024 - Accepted on, 29 November 2024 - Published on, 08 December 2024

ABSTRACT

In the vast majority of low-and middle-income countries, performance of health systems continues to be abysmally poor with unacceptably low health outcomes. This is not unconnected with implementation of evidence-deficient health policies. Critical research evidence contributes to strengthening health policies to ensure clear cut targets and context specifics that adequately address identified health challenges and inequities. This study modeled a computing paradigm for brokering knowledge translation process and assisting health policymakers in promoting evidenced-informed policymaking. It strategically evaluates and assesses levels of evidence content and predicts implementation prospects of health policy documents. Its development process adopted object-oriented methodology for structural analysis and design specifications. Visual Basic.net and standard query language server were deployed at the front-end and back-end implementation processes, respectively. The study designed an algorithm based on discrete choice experiment technique in an iterative four-scaled user-defined parametric options for rating policy features and assessment of overall policy prospect. Salient policy features/attributes were assembled as assessable variable entities. It adapted machine learning linear model to classify attributes into 6-domains to reflect the WHO promoted 6-policy cycle of a health system. Aggregated scores of policy features across all domains are utilized to compute policy overall grade-point in percentage weight. PROPHET was used to assess thirty-three (33) national health policies extracted from online repository warehousing health policy documents in Nigeria known as *policy information platform*. The result shows that only 11 out of the 33 (33.3%) policies passed with at least 50% grade-point fixed in this study as minimum benchmark for implementation considerations. This system rates policy features, assesses overall implementation prospect of policies with seamless real-time data validation and referencing across modules. PROPHET is expected to aid health policymakers in amplifying evidence-informed policymaking for improved health outcomes.

Index words: Evidence-informed, Grade-point, knowledge translation, Object-oriented, Policy feature/attribute, Policymaking, User-defined.

I. INTRODUCTION

Provision of appropriate policy direction especially in the health sector is critical to the realization of Sustainable Development Goals (SDGs) in health and could further be strengthened with innovative computer-aided tools. Lack of relevant computing technology aid for assessing the research evidence content and classified ways to validating prospects of policies adversely affect health outcomes. The strengthening of global health systems has been facilitated through rapid utilization of the ever-evolving computer technological processes [1-3]. Computing and information technologies (CITs) has sustained their growing trends and tremendous impacts on the global health systems in terms of improved dissemination of public health information, facilitating public discourse around policy related issues and dialogue around major public health threats [3-5].

There is an increasing global acceptance that one major way to address weak health systems and improve health outcomes in low-income settings is by the development and implementation of health policies that are evidence-informed [6-8]. Thus, the strengthening of low- and middle-income countries' (LMICs') health systems would require strong drive anchored on the use of research evidence in formulating health policies in compliance with systems' thinking perspectives [9,10]. Systems' thinking ensures equitable schedule of resources across the six health systems' building blocks, in order to maintain undisrupted balance among the various domains and forestall all forms of imbalance and inadequacies. For instance, the imbalance in cost of healthcare services negatively impacts on overall uptake especially among the poor and vulnerable groups for obvious reasons [11]. Recognizing the importance of utilizing best available research evidence in health policymaking by policymakers in Nigeria is still at a low level leading to the formulation of policies based on assumptions [9,12-13]. Implementation of such policies have in most cases resulted in an effort in the futility and waste of scarce resources as the policies never achieved their purpose.

Evidence-informed policymaking (EIP) is a critical process involving proven scientific methods that creates avenues where researchers are linked with policymakers for active collaboration [14]. This ensures integration of evidence-based interventions with community preferences to improve and balance policymaking initiatives [15,16]. Evidence-informed policy process has rationalist assumptions that health policies should ultimately be based on evidence from research [17]. A study suggests positive effect of adopting electronic technology support systems to harness and synthesize varieties of evidence for sustainable policy practice [18,19]. It leverages the use of electronic driven interventions in the government and overall management of public health data/information and health systems operations [20,21]. Such approach suffices knowledge translation process for harvesting experts' views and experience as useful resources that can cumulatively build policy support [22-24]. In other words, evidence-to-policy link can be energized more by maximizing the reach potentials of emerging pertinent technologies to pull-through policymaker's capacity.

Advances in computing techniques are rapidly creating the framework upon which almost everything works, even transforming and reforming the trends of

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

health-based undertakings by creating sensitive tools for hospitals, pathology laboratories and dental clinics, etc. [3,25]. They provide strategic platforms with metric-engineered processes for real-time health information management and control [26,27]. Such systems may include the development of digital or mobile health solutions for improved knowledge dissemination, community participation in prevention programs, advocacy and policy dialogues for effective public health [28,29]. Digital health technology has provided a more efficient, accessible, and effective means to collect, analyze, store and share health data [30]. Just as computerized systems now permeate complex managerial areas in industries for evaluating proposed acquisitions, this new system models a window of applicability to ease-off problem-solving approaches associated with health policymaking [31,32]. Expert's ideas weaves solutions by creating enhanced information access and flow process, with interactive sharing and unique exchange technique [33,34]. These avails policymakers' seamless opportunities to explore and utilize unlimited evidential resources on policy-driven issues especially those of the health sector. There exist several information systems platforms such as medical transcription tool for physicians and healthcare providers to leverage on, in treatment and care for patients [35,36]. These tools facilitate interpretation of handwritten prescriptions, updates medical case histories, along with emerging trends in network technologies that connect sensors and input devices in patient home to a "home-health-care provider" made home care for even gravely ill patients a possibility [37-39].

Several action frameworks and multifaceted approaches have been developed to identify pertinent domains and guide development of organizational tools and systems that may facilitate research use by policymakers. Such action frameworks include: (a) **The SPIRIT Action Framework,** which is an Intervention Trial that is a structured approach to selecting and testing strategies to increase the use of research in policy [40]. (b) **The SAGE Framework,** which is a tool to evaluate how policymakers engage with and use research in health policymaking [41]. (c) **The ORACLe framework,** which is a comprehensive system to measure and score organizations capacity to engage with and use evidence from research in health policymaking process [42]. (d) **The SEER framework,** which is designed to determine the well validated measures to identify priorities for capacity building in engaging with research outcomes and researchers [43]. These previous action frameworks inspired the development of a computer-driven approach and conceptual framework known as the PROPHET *(Policy Research-evidence Organizer and Public Health-policy Evaluation Tool)*. The PROPHET is designed to facilitate standards, support improved precision in decision making and abate waste of scarce resource through strategic engagement of evidence informed policymaking process. The PROPHET is therefore a computer software paradigm, which is to serve as a tool to aid knowledge translation process. It was intended that the software would assist policymakers assess multiple aspects of health policy documents compliance with evidence-to-policy perspectives. The PROPHET is also intended to predict the prospects of a given policy successfully achieving its purpose prior to implementation, thereby suggesting the feasibility of the policy option addressing targeted areas of needs.

A. CONCEPTUAL FRAMEWORK OF THE PROPHET

Figure 1 describes the high-level model (HLM) architecture of the PROPHET developed to showcase the entire system immediately by specifying the basic activities and attributes associated with it. It is an architectural model configured to identify and describe data elements with basic functional components and the logics infused into the system and synchronized for efficient data communication traffic flow.

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

The PROPHET's operation is designed to revolve around the central strand of the system labeled "Interface Control Valve" configured into four-fold activity modules. On the left stroll is the "Input Switch Function" that comprises Login Reset, Add Domain, Add Features and create Option-list. These are initial basic system tools exclusively engaged by an administrator (Admin) for classical operations and subject to necessary modifications whenever it is called for. From the right side of the HLM is the "Domain Classification Framework" which defines and establishes the six domains encasing all the profiled policy features/attributes. Directly underneath the interface control valve is the "Computation Paradigm" and the "Report Generation" modules. The Computation Paradigm undertakes registration of policy documents, rating of its features as appropriately specified and performing overall policy assessment, to determine its percentage weighted grade-point (PWGp). On the other hand, the Report Generation keeps track of, stores and seamlessly recalls where necessary, all the records of activities or transactions traversing the entire system. Figure 1 presents high-level model architecture for the system (PROPHET).

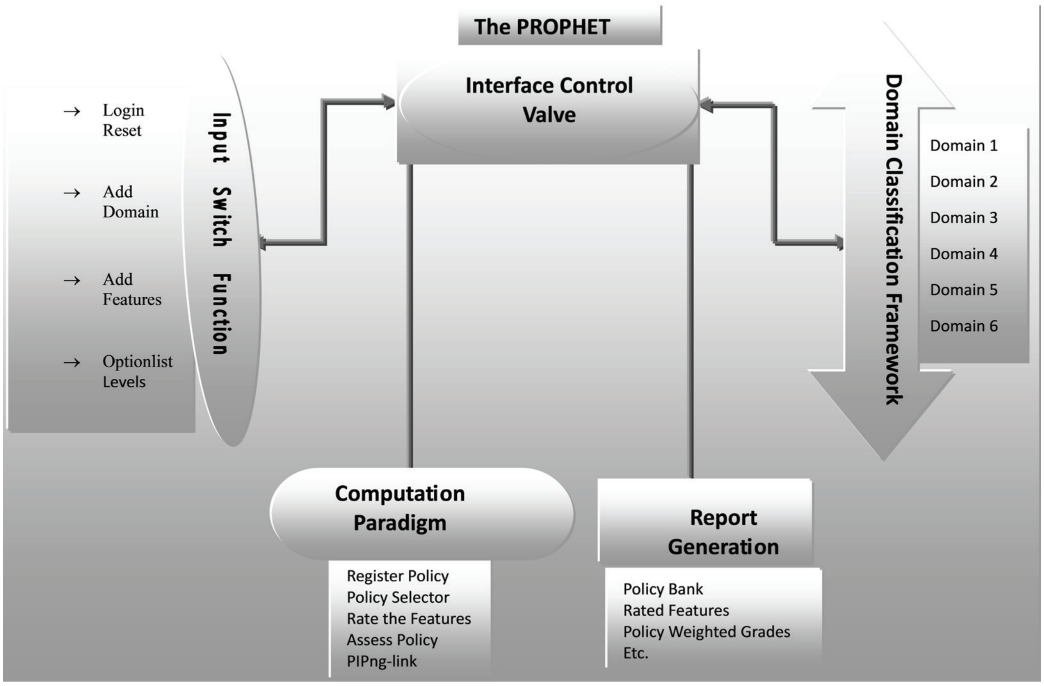


Fig. 1. High-level model architecture for the system (PROPHET)

II. MATERIALS AND METHODS

Software engineering development process broadly encompasses three major phases viz: systems analysis, design and implementation [44-46]. Undertaking these basic activities requires adoption of appropriate methodology and in the development of the PROPHET, the Object-Oriented methodology (OOM) was considered suitable and adopted. This is due to the fact that OOM ensures concise definition of the problem and ease of exploring definite concepts associated with the problem domain [47][48]. The OOM helps to model the system in a way to easily manipulate object pieces for proper interaction and generation of events among functional components, as well as ensure adequate structural data entity representation among various system components [47]. The relevant OOM tools and materials utilized for the PROPHET development analysis and design includes: flow charts, sequence diagram, use-case diagram and activity diagram. These OOM tools have been shown to facilitate systems analysis and design process of software development that describe the interactive flow of operations within and/or among the system components [47,48]. The PROPHET analysis and design are critical steps that involve a systematic way of x-raying the structural and logical processes that underpin the various operations. It indicated phase-by-phase and module-by-module development process [44-46].

A. DESIGN AND ANALYSIS OF PROPHET FLOWCHART OPERATION

To transform the conceptual framework of the PROPHET into an operational system, the overall system flowchart was designed as illustrated in figure 2 as an OOM generic tool for analysis and design [47,48]. The traffic flow operation was designed such that at the onset, the system access point is launched to select a definite level of usage either as "Admin" or "Ordinary policymaker". In any case, the user needs to enter a unique username and password to login and have access to the system's main user interface. In another case, to have access to the system main user interface, the system was designed such that the user has to enter a unique username and password to login. The system was also designed such that the Admin can create or modify basic functions of the system such as adding/deleting domains, policy features, rating levels etc., and can perform other sub-operations which include enrolling or profiling a policy document (A), rating its features (B) and assessing the policy document (C). On the other hand, the system was also designed such that a policymaker can view already assessed health policy document to guide in implementation decision and can pick up a registered policy, rate its features (B) and assess the policy (C) to determine its overall weighted grade-point. These operations were designed to be automatically validated from within the system, systematically and routinely ensuring error-free transmissions. Figure 2 shows system development flowchart operation.

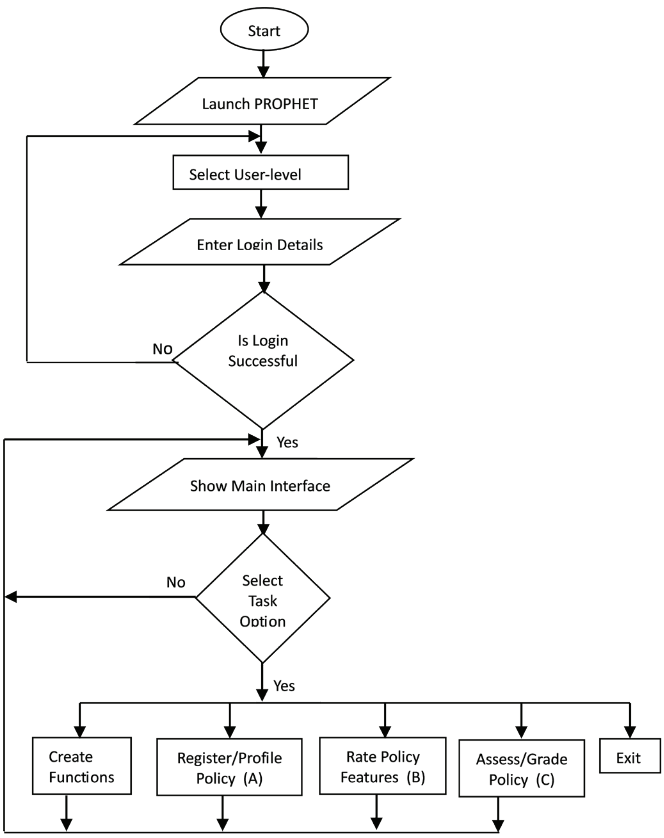


Fig. 2. System development flowchart operation

B. DESIGN AND ANALYSIS OF PROPHET SEQUENCE DIAGRAM OPERATION

The PROPHET was designed such that there are sequential interactions between and amongst actors and objects in the system. In figure 3, sequence diagram as a pertinent OOM tool was designed to depict and analyze interactive operations in the PROPHET where an admin or a policymaker could trigger-off an action within the system via “Select Policy” item at the instance of policy information object class, which supplies the policy identifier (PID) item. The PROPHET was further designed with “Activate System” command to extract and return required information of selected policy document in the “Policy Bank” and enables activation order “Activate RF” for a user to commence a definite rating task. If a policy document is not validly selected, activation order would fail and fresh request is retransmitted. Once the transmission service is upheld and an activation order is validated, the PROPHET was designed to load the domain selection Form for commencement of a domain-by-domain rating of the features/attributes with the “Activate RF” command enabled. The next action stage is where the PROPHET was designed to display a status update on a message box indicating the cumulative value accruing from rate features “RF” operation, and this value was used at the next action stage to compute overall policy weighted grade-point with details from update file tool. In assessing a policy's weighted grade-point (PWGp), the PROPHET was designed to call-up detailed updated value from “Authorize RF” through the “summarize policy identifier (PID)” command tools. With a click event of grade-point command button “Grdp summary”, the PROPHET was designed to automatically activate PWGp computation order and display the overall result on the message box. Thereafter, the PROPHET was designed to allow user perform an end-to-end validation action by selecting submit/update database activation order and click exit to quit that round of operation. Figure 3 shows the sequence diagram operation for the system (PROPHET).

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

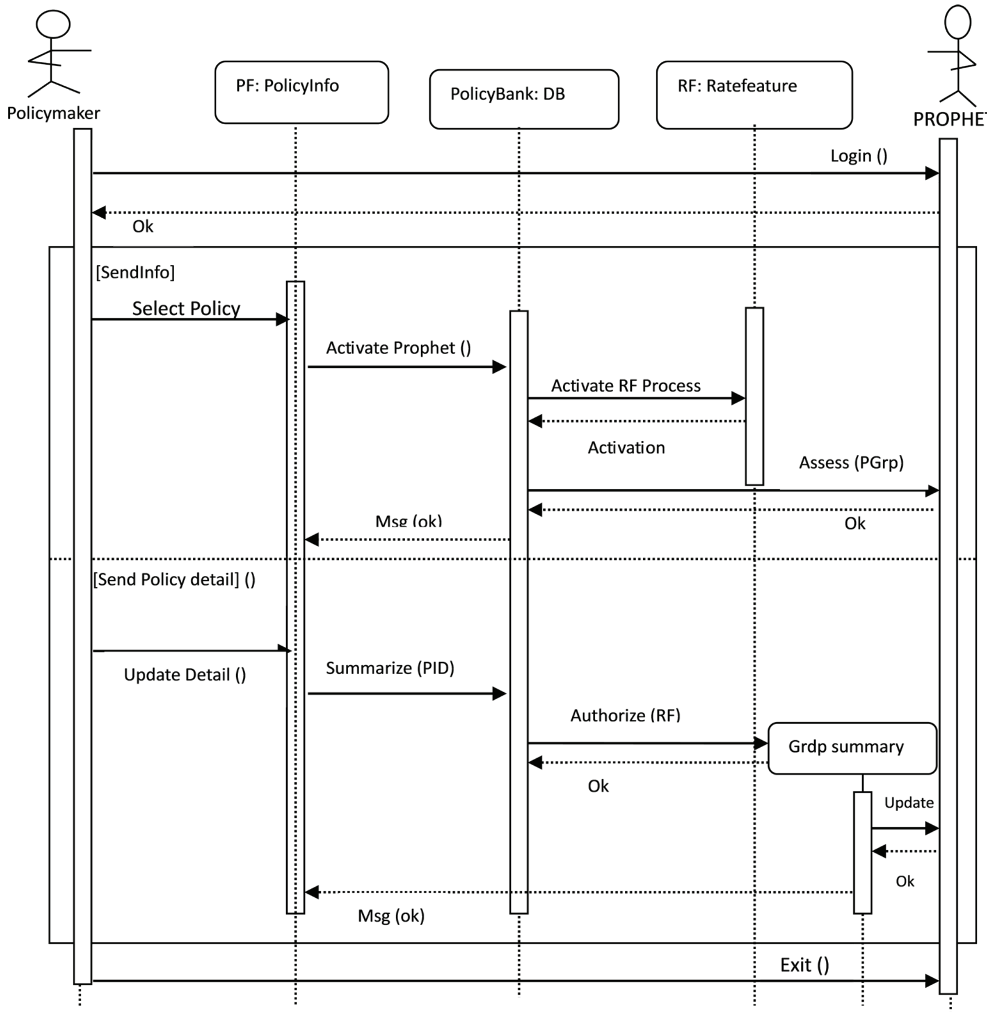


Fig. 3. Sequence diagram operation for the system (PROPHET)

C. DESIGN OF PROPHET DOMAIN CLASSIFICATION

This is the PROPHET development phase that saw to the design and formulation of the six (6) operational domains of the system. In developing the PROPHET, the researchers designed a six-domain framework and classification model adapted in reflection of the health policy cycle development process of the World Health Organization [13,49-51]. This was used to facilitate a systematic classification of all the identified salient policy features/attributes into suitable domain as assessable variable entities according to their functional relevance. All these policy features/ attributes represent the input functions designated in PROPHET as active data traversing across its various related data fields. In other words, the PROPHET was designed to adapt the structured six policy cycle development process as a classified ordered framework of six (6) key domains encasing all salient policy features/attributes [13,49-51]. Each domain encases an array of relatively peculiar policy features or attributes with requisite information metric critical to policymaking process. Overall, the PROPHET was designed to enumerate thirty-six (36) default policy features/attributes which form the basis for assessing profiled health policy documents. The PROPHET was designed with a dynamic nature that allows for modifications in both ratable features and the rating option-list levels. These domains with their various policy features or attributes, are represented and explicitly defined in table I. This table indicates the content adequacy consideration of each domain, which can be modified as future research findings provides more useful evidence per subject matter.

TABLE I
CLASSIFICATION OF DOMAINS AND CONTENT DEFINITIONS

Domain Code	Domain Title	Domain Features/Attributes
D1	Preliminary Concepts	comprises all standardized preliminary components of a standard policy document which includes: policy title, source of policy, date of production, preface, foreword, acknowledgement, contributors/ stakeholders, acronyms/glossary, policy lifespan, abstract/executive summary, roles/responsibilities, annexure and reference/ bibliography
D2	Introductory & Issue Raising	comprises all the introductory factors that shades light on the cause(s) of study and benefits. They include background concepts, policy statement/problem definition, policy goals and objectives, policy scope and policy justification/rationale
D3	Scientific Process & Policy Design	It comprises all scientific process and technical approach considerations adopted to guide policy production decisions. This includes methodology, policy framework, priorities/priority areas, policy guideline, research evidence, analyzing roles of actors/institutions ideologies, **policy situation analysis and policy institutional context analysis**
D4	Public Support	this entails the rigorous efforts made to sell the policy to the people (people-oriented and driven) – ensures it does not impede on their norms and values. These are: policy dialogue/consensus building, advocacy drive and policy recommendation
D5	Legislative Decision & Policy Support	it comprises all the necessary steps taken to bring a policy into legitimate force – making it binding on all the target population and coverage areas. They include policy legislation, legal and regulatory framework
D6	Policy Implementation	this refers to the critical step-by-step process followed to appropriately provide and deploy adequate resources needed to enforce realization of overall policy goals and objectives. These includes budgeting issues, policy implementation strategies/planning/guidance, policy monitoring and evaluation, supervision mechanism, policy dissemination strategies, and communication/social mobilization.

D. DESIGN OF RATING OPTION-LIST LEVELS

In developing PROPHET, the rating option-list levels was designed as a parameter for rating policy features by exploring the dynamics of discrete choice experiment technique to formulate an iterative user-defined algorithm in a four-scaled option list [52-54]. This framework was designed to evaluate the level of availability or content adequacy of designated input variable in each domain so as to duly assign appropriate scale value indicated in table II. Due to possible changes in the input variables, PROPHET designed a mathematical and regression model wherein output values would be expressed as the linear combination of a set of input variables [55,56]. This implies that:

$$y = w1x1 + w2x2 + w3x3 \tag{1}$$

where y describes a residual function, w1…w3 is the weight value assigned to the input variables and x1…x2 is the probable error function.

The user-defined parametric discrete variable weighs policy features/attributes over the four scaling levels for purposes of rating each variable into a definite value. This logical and iterative structural process systematically rates these variables and records their scores on domain-by-domain basis. Precisely, table II illustrates the four-scale rating option lists designed for this system development with their corresponding discrete values are specified thus: not included "0", unsure "1", partially included "2" and fully included "3".

TABLE II
THE PROPHET OPTION-LIST LEVELS DEFINITION

Option-List	Rating Value	Level Definition
Not Included	"0"	It means a feature/attribute is inadvertently unavailable and was not represented or considered in any form in a policy document
Unsure	"1"	A feature is not categorically stated or included, instead it has a grossly inadequate description relative or similar to it captured as part of a policy document
Partially Included	"2"	Some information about an attribute or a feature is included in a policy document but not concisely described (content inadequacy)
Fully Included	"3"	An attribute or feature is both available, concisely described and all relevant details are exclusively represented in the policy (content adequately)

E. DESIGN OF POLICY RATING AND ASSESSMENT ALGORITHM

The system (PROPHET) algorithm was designed to collate cumulative scores or values of rated features (RF) within domains and all summarized into a domain weighted aggregate score. Policy rating process is designed to valuate both item availability and content adequacy in line with established discrete option levels. The values allotted to rating options duly selected will linearly add up to generate total score for that domain. In other words, the summative score accruable from a selected domain was designed to be calculated through equation 2, as the linear combination of the input variables (rated features) within that domain as expressed thus:

$$DWV = \Sigma\,(RF_1 + RF_2 \ldots + RF_{n-6}) \qquad (2)$$

Where;
DWV: represents domain weighted value, which is meant to hold the total result of sum of all the values arising from the individual rated features of a policy.
RF = this stands for rated feature and it represents each of the features in a domain which is selected (activating) before clicking an option-list suitable for its availability or level of adequacy in the policy being assessed. The numeric value of the option chosen is recorded in favor of that feature and used to calculate total domain score (DWV).

On the other hand, the PROPHET was designed to calculate cumulative policy weighted grade-point (PWGp) in percentage terms. This is done by collating the outcome of domain weighted value (equation 2) across all six domains using the algorithm in equation 3, expressed as the linear combination of all domain weighted values contained in equation (2) above and derived thus:

$$PWG_p = \Sigma\,(DWV_{n1} + DWV_{n2} \ldots + DWV_{n-6})\ /MAS * 100 \qquad (3)$$

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

PWGp = it stands for Policy Weighted Grade-point in percentage. This is the PROPHET final policy assessment process carried out to determine potential or prospective implementation strength of a policy measured in percentage (%) weight factor.
DWV = this represents Domain Weighted Value, as collated from across the six (6) domains in equation (2) after rating the policy features.
MAS = stands for Maximum Accruable Score across all domains. It is obtained by calculating the total number of registered policy features and multiplying same with the highest possible numeric option-list value established in the system (i.e. "3"). It implies that MAS is the total number of registered policy features multiplied by highest possible value "3". Aggregated score from policy features across all domains is utilized to compute policy overall grade-point in percentage weighting.

III. RESULTS

In the developed system called PROPHET, the input and output objects that were designed in PROPHET were implemented and tested to ascertain their functionalities following software engineering development routine. The implementation routine was as provided in the integrated development environment (IDE) of Visual Basic. net at the front end, the standard query language (SQL) Server at the back end and a relational database called "PolicyMakingDB". The inherent logics resulted in the development of this novel piece called PROPHET, technically configured with main-menu interface that integrated and activated various modules of the PROPHET for seamless real-time functionality.

A. MAIN MENU INTERFACE MODULE

The PROPHET activities were implemented on a well-developed window-based and highly interactive graphical user-interface, which serves as the main menu for access. Figure 4 presents the PROPHET main menu interface made up of robust controls that insulates users from underlying technological tendencies which ensures operational flexibility. It provides simplified and easy independent access to all the controls/menus for initiating and undertaking a policy assessment task with timely response to errors. There are four menu items that characterized the main interface as follows: Admin, Task, Report and About. These menu items and other submenus were meant to handle all operation beginning with policy profiling to the overall policy assessment exercise. Figure 4 illustrates the screenshot of main menu interface window.

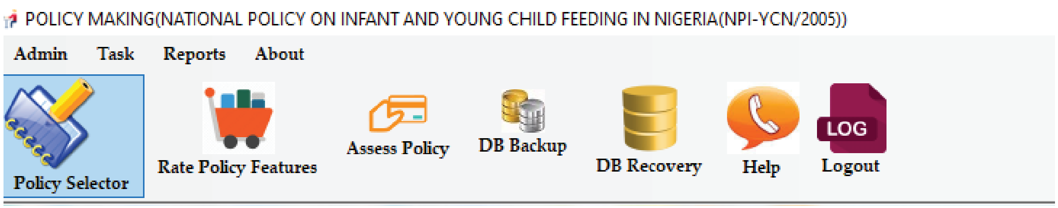


Fig. 4. Screenshot of main menu interface window

B. MODULE FOR RATING POLICY FEATURES

The PROPHET implemented policy rating activity through the development of the model of four-scale option-list level shown in figure 5. The model comprised of the following rating levels: "not included", "unsure", "partially included" and "fully included" respectively assigned with numerical values "0", "1", "2" and "3" for each

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

level. The ratable policy features of policy documents were designed into a set of number knobs (1, 2, 3, ...n) lined chronologically at the base of the rating window underneath the option-list levels. Each feature or attribute is comparatively weighed and pined with a commensurate numerical value, akin to the extent or its availability strength in the policy being assessed. The PROPHET undertakes the rating activity procedurally with the aid of pre-determined logical activations leading to domain-after-domain and attribute-after-attribute operations. Figure 5 show the screenshot of dialog window for rating policy features/attributes.

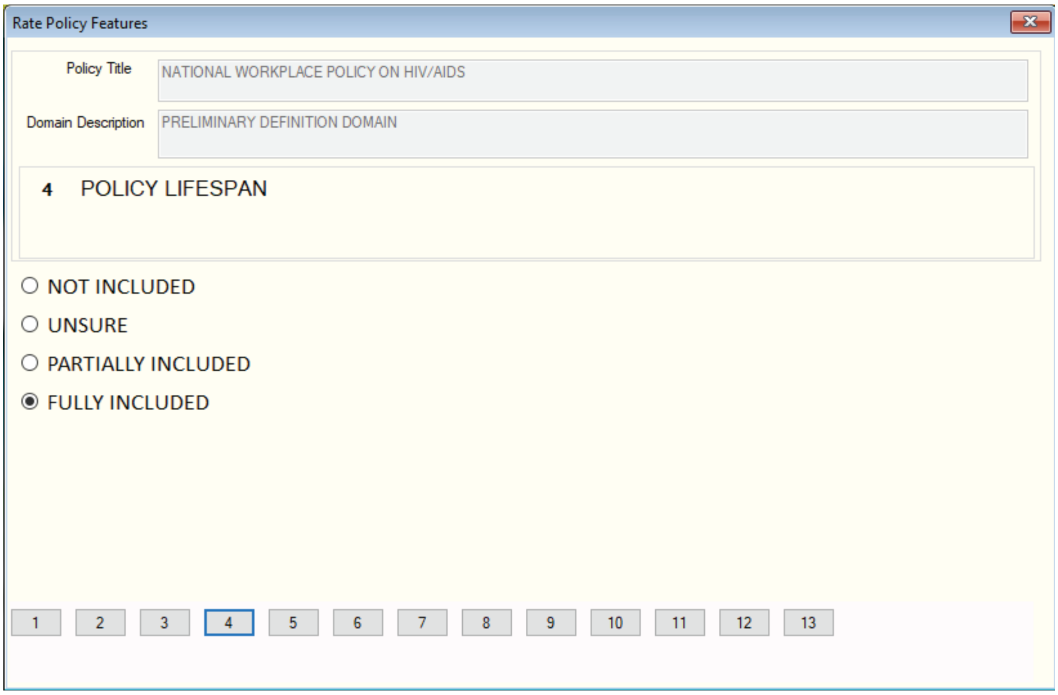


Fig. 5. Screenshot of dialog window for rating policy features/attributes

C. MODULE FOR COMPUTING POLICY GRADE-POINT

At this module, figure 6 was a dialog window for the PROPHET used to initiate and coordinate the processes of computing and determining overall policy weighted grade-point (PWGp) in percentage scale. This operation is carried out by selecting/ clicking the assess policy menu item to load the "Compute Policy" window. Next step is to select/click "All Domain" option button to activate the rated values of the six (6) domains in readiness for next operation. This action displays cumulative summary value in the Total Domain Score box. Thereafter, select/click the command button bearing "Process Gradepoint" to get the policy weighted gradepoint (PWGp) and then tick the "I Agree" option-box before finally clicking the "Submit" command button to transmit into the database and conclude the operation. Figure 6 presents the screenshot of dialog window for policy grade-point assessment.

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

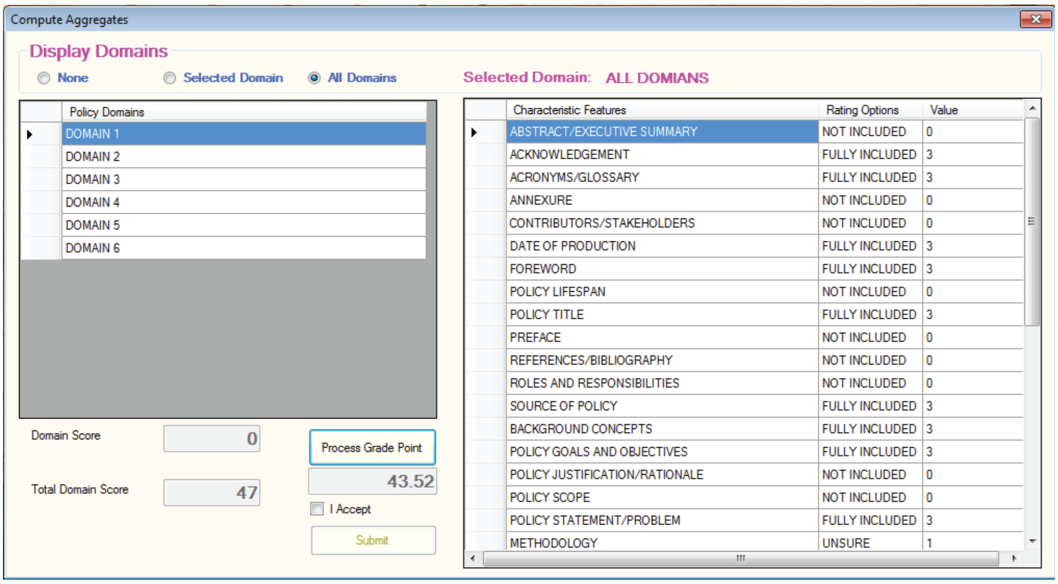


Fig.6. Screenshot of dialog window for policy grade-point assessment

D. SUMMARIZED RESULT OF HEALTH POLICY DOCUMENTS ASSESSED WITH PROPHET TOOL

Table III captures the results of the thirty-three (33) policy documents that were assessed with their corresponding weighted grade-point using the PROPHET software tool. From the results, only eleven (11) policies marked with green color out of a total of thirty-three (33) policies scored up to the minimum benchmark of fifty (50) percent grade-point and above. That is for serial numbers 1, 4, 7, 13, 14, 20, 22, 24, 27, 28 and 29. Whereas the other twenty-two failed short of the designated fifty percent study minimum benchmark.

TABLE III
SUMMERY OF RESULTS OF POLICIES ASSESSED USING THE PROPHET

SN	YR. CODE	TITLE OF POLICY	ORIGIN	SCORE	PWGp
1	DEN-PF/2010	A GENDER POLICY FOR THE NIGERIA POLICE FORCE	NPF	60	55.56
2	MAL-FIL/2013	GUIDELINES FOR MALARIA-LYMPHATIC FILARIASIS CO-IMPLEMENTATION IN NIGERIA	FMoH	46	42.59
3	IPH-G/2013	INTEGRATING PRIMARY HEALTH CARE GOVERNANCE IN NIGERIA	NPHCDA	39	36.11
4	MS-PHC/NON	MINIMUM STANDARDS FOR PRIMARY HEALTH CARE IN NIGERIA	NPHCDA	62	57.41
5	MAL-T/2005	NATIONAL ANTIMALARIAL TREATMENT POLICY	FMoH	47	43.52
6	NCH-P/2006	NATIONAL CHILD HEALTH POLICY	FMoH	53	49.07
7	NDC-MP/2015	NATIONAL DRUG CONTROL MASTER PLAN	NDLEA	71	65.74
8	ND-P/2005	NATIONAL DRUG POLICY	FMoH	34	31.48
9	NFP-RHSP/2009	NATIONAL FAMILY PLANNING/REPRODUCTIVE HEALTH - SERVICE PROTOCOLS	FMoH	35	32.41
10	NGP-SF/2008-2013	NATIONAL GENDER POLICY STRATEGIC FRAMEWORK (IMPLEMENTATION PLAN)	FMoH	49	45.37

11	MAL-T/2011	NATIONAL GUIDELINES FOR DIAGNOSIS AND TREATMENT OF MALARIA	FMoH	39	36.11
12	HIV-TR/2010	NATIONAL GUIDELINES FOR HIV AND AIDS TREATMENT AND CARE IN ADOLESCENTS AND ADULTS	FMoH	38	35.19
13	NG-PAE/2007	NATIONAL GUIDELINES FOR PAEDIATRIC HIV AND AIDS TREATMENT AND CARE	FMoH	54	50
14	NG-PMT/2010	NATIONAL GUIDELINES FOR PREVENTION OF MOTHER-TO-CHILD TRANSMISSION OF HIV	FMoH	56	51.85
15	NH-PP/	NATIONAL HEALTH PROMOTION POLICY	FMoH	46	42.59
16	NIP/2009	NATIONAL IMMUNIZATION POLICY	NPHCDA	45	41.67
17	NNG-NCD/2014	NATIONAL NUTRITIONAL GUIDELINE ON NON-COMMUNICABLE DISEASE PREVENTION, CONTROL AND MANAGEMENT	FMoH	42	38.89
18	NPA-FN/2002	NATIONAL PLAN OF ACTION ON FOOD AND NUTRUITION IN NIGERIA	NPC	50	46.3
19	NP-FNN/2001	NATIONAL POLICY ON FOOD AND NUTRUITION IN NIGERIA	NPC	38	35.19
20	NP-HIV/2003	NATIONAL POLICY ON HIV/AIDS	FMoH	55	50.93
21	NP-IYC/2005	NATIONAL POLICY ON INFANT AND YOUNG CHILD FEEDING IN NIGERIA	FMoH	52	48.15
22	NPM-DT/2011	NATIONAL POLICY ON MALARIA DIAGNOSIS AND TREATMENT	FMoH	56	51.85
23	NP-PPH/2005	NATIONAL POLICY ON PUBLIC PRIVATE PARTNERSHIP FOR HEALTH IN NIGERIA	FMoH	45	41.67
24	NP-HDAY/2007	NATIONAL POLICY ON THE HEALTH & DEVELOPMENT OF ADOLESCENTS & YOUNG PEOPLE IN NIGERIA	FMoH	57	52.78
25	NRH-PS/2001	NATIONAL REPRODUCTIVE HEALTH POLICY AND STRATEGY	FMoH	53	49.07
26	NRH-SF/2002	NATIONAL REPRODUCTIVE HEALTH STRATEGIC FRAMEWORK AND PLAN	FMoH	49	45.37
27	NSH-P/2006	NATIONAL SCHOOL HEALTH POLICY	FMoE	56	51.85
28	NS-GHC/2005	NATIONAL STRATEGIES AND GUIDELINES FOR HOME AND COMMUNITY MANAGEMENT OF MALARIA	FMoH	63	58.33
29	NN-BF/2005	NIGERIA NATIONAL BIOSAFETY FRAMEWORKS	FMoEnv.	64	59.26
30	POLIO/2012	NIGERIA POLIO ERADICATION EMERGENCY PLAN	NPHCDA	39	36.11
31	TASK/2014	TASK-SHIFTING AND TASK-SHARING POLICY FOR ESSENTIAL HEALTH CARE SERVICES IN NIGERIA	FMoH	43	39.81
32	NBP/2006	NATIONAL BLOOD POLICY	NBTS	38	35.19
33	NG-TB/2008	NATIONAL GUIDELINES FOR TB INFECTION CONTROL	FMoH	41	37.96

NBTS national blood transfusion service, NPC national population commission

IV. DISCUSSION

The study described an iterative technical process for strategically assessing quality of research evidence utilization and content validation in a policy document. PROPHET software was developed and used to establish the fact that computer-driven activities apply in virtually all fields of human endeavor, both to facilitate standards, support improved precision in decision making and abate waste of scarce resource through strategic conceptual approaches [38,57-59]. The policy assessment result in table III has only eleven (11) policies scoring up to fifty percent, which clearly suggests that there are very few policy documents that were made with sufficient research evidence content. This underscores the need for policymakers to adopt technology driven approaches such as the PROPHET capable of facilitating processes in compliance with evidence-to-policy perspective.

The outcome of the assessment of the thirty-three (33) policy documents that was done using the PROPHET showed its critical role in advancing evidence informed policymaking and implementation. Two-third of the policy documents assessed were shown to have low weighted grade-points, signifying that they were formulated without adequate use of research evidence and as such would be difficult to implement and would not yield intended benefits. One can deduce from this outcome, that if as much as two-third of the national policy showed this outcome, then many of the suggested policies would fall into this category and therefore need urgent reviews. Such poor outcome is in keeping with a recent report of suboptimal use of research evidence in policymaking [60]. This can be explained by the existence of weak and sometimes lack of researcher-policymaker linkages and platforms [61,62]. Understandably, with such gaps, policymakers would tend to use routinely collected data rather than research evidence from external academic institutions, as reported in a study [60].

V. CONCLUSIONS

The development of Policy Research-evidence Organizer and Public Health-policy Evaluation Tool (PROPHET) was successful, and it represents a new trend in evidence-informed policymaking (EIP) perspectives. It is a flexible, efficient and user-friendly interactive piece of software that allows for relevant context-specific modifications in conformity with any definite health issue. PROPHET has been tested for functional effectiveness with health policy document extracted from an online repository warehousing all health policy documents in Nigeria known as *policy information platform*. The test result shows that only 11 out of the 33 policies passed with at least 50% grade-point reputed in this study as minimum benchmark for implementation or be referred for necessary review. It rates policy features, assesses overall implementation prospect of policies with seamless real-time data validation and referencing across modules. PROPHET is expected to aid public health policymakers in amplifying *evidence-informed policymaking* for improved health outcomes. The researchers recommend government agencies adoption of this novel tool in facilitating compliance with the ideals of systems' thinking (evidence-to-policy perspectives) encased in the 6-building blocks of the health systems.

ACKNOWLEDGEMENT

The researchers would like to acknowledge and thank the leadership of the African Institute for Health Policy and Health Systems for facilitating processes to accessing the policy documents used in this study through the *policy information platform*

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

(www.pipnigeria.org). The support of the Ebonyi State Coordinator of the National Health Policy Plus (HP+) programme in terms of supplying relevant information is equally well acknowledged.

FOOTNOTES

CONFLICT OF INTEREST: The authors declare that they have no competing interest.

ETHICAL STATEMENT: The authors are accountable for all aspects of this study in ensuring that all questions relating to the accuracy or integrity of any part of the study are appropriately investigated and resolved.

VI. REFERENCES

[1] Mbam BCE, "Information Technology and Management Information System," *Our Saviour Press LTD, Enugu*, pp. 132–133, 2002.

[2] S. Pokhrel, "Scaling up health interventions in resource-poor countries: What role does research in stated-preference framework play?," 2006. doi: 10.1186/1478-4505-4-4.

[3] D. S. M. A. Burney, N. Mahmood, and Z. Abbas, "Information and Communication Technology in Healthcare Management Systems: Prospects for Developing Countries," *Int J Comput Appl*, vol. 4, no. 2, 2010, doi: 10.5120/801-1138.

[4] A. Chetley *et al.*, "Improving health, connecting people: the role of ICTs in the health sector of developing countries A framework paper," *Infodev*, no. 7, 2006.

[5] K. C. Laudon and J. P. Laudon, *Management information systems – Managing the digital firm*, Eleventh Edition. New Jersey: Pearson Education Inc, 2010.

[6] Z. Shroff, B. Aulakh, L. Gilson, I. A. Agyepong, F. El-Jardali, and A. Ghaffar, "Incorporating research evidence into decision-making processes: Researcher and decision-maker perceptions from five low- and middle-income countries," *Health Res Policy Syst*, vol. 13, no. 1, 2015, doi: 10.1186/s12961-015-0059-y.

[7] C. J. Uneke, A. E. Ezeoha, H. C. Uro-Chukwu, C. T. Ezeonu, and J. Igboji, "Promoting researchers and policy-makers collaboration in evidence-informed policy-making in Nigeria: Outcome of a two-way secondment model between university and health ministry," *Int J Health Policy Manag*, vol. 7, no. 6, 2018, doi: 10.15171/ijhpm.2017.123.

[8] V. Tudisca *et al.*, "Development of measurable indicators to enhance public health evidence-informed policy-making," *Health Res Policy Syst*, vol. 16, no. 1, 2018, doi: 10.1186/s12961-018-0323-z.

[9] C. J. Uneke, A. E. Ezeoha, C. D. Ndukwe, P. G. Oyibo, and F. Onwe, "Enhancing health policymakers' capacity to use information and communication technology in Nigeria," *J Health Inform Dev Ctries*, 2011.

[10] Sherri Rose, "Machine Learning in Biostatistics & Health Policy," PhD dissertation, Harvard Medical School, 2016.

[11] I. S. Fall *et al.*, "Integrated Disease Surveillance and Response (IDSR) strategy: Current status, challenges and perspectives for the future in Africa," 2019. doi: 10.1136/bmjgh-2019-001427.

[12] "Supporting Policy In health with Research: an Intervention Trial (SPIRIT)—protocol for a stepped wedge trial," *BMJ Open*, vol. 4, no. 7, p. e005293, Jul. 2014, doi: 10.1136/bmjopen-2014-005293.

[13] J.-P. Rodrigue, C. Comtois, and B. Slack, *The Geography of Transport Systems.* Routledge, 2016. doi: 10.4324/9781315618159.

[14] B. W. Head, "Toward More 'Evidence-Informed' Policy Making," *Public Adm Rev*, vol. 74, no. 4, pp. 72–84, May 2019, doi: 10.1111/puar.12475.

[15] R. Larocca, J. Yost, M. Dobbins, D. Ciliska, and M. Butt, "The effectiveness of knowledge translation strategies used in public health: A systematic review," 2012. doi: 10.1186/1471-2458-12-751.

[16] F. El-Jardali, J. Lavis, K. Moat, T. Pantoja, and N. Ataya, "Capturing lessons learned from evidence-to-policy initiatives through structured reflection," *Health Res Policy Syst*, vol. 12, no. 1, 2014, doi: 10.1186/1478-4505-12-2.

[17] S. Ettelt, B. Hawkins, and A. Alvarez-Rosete, "Working Paper # 3 Analysing evidence use in national health policy-making – an institutional approach," *London School of Hygiene and Tropical Medicine GRIP-Health Programme*, 2013.

[18] I. Ajah and H. C. Inyama, "A Model of DNS-Based Bank Credit Risk Management," *ARPN Journal of Systems and Software*, vol. 3, no. 6, pp. 106–128, 2013.

[19] S. A. Abdulrahman and K. Ganasegeran, "M-health in public health practice: A constellation of current evidence," in *Telemedicine Technologies: Big Data, Deep Learning, Robotics, Mobile and Remote Applications for Global Healthcare*, 2019. doi: 10.1016/B978-0-12-816948-3.00011-8.

[20] N. Kasoju *et al.*, "Digital health: trends, opportunities and challenges in medical devices, pharma and bio-technology," *CSI Transactions on ICT*, vol. 11, no. 1, 2023, doi: 10.1007/s40012-023-00380-3.

[21] B. Ebenso *et al.*, "Impact of using eHealth tools to extend health services to rural areas of Nigeria: Protocol for a mixed-method, non-randomised cluster trial," *BMJ Open*, vol. 8, no. 10, 2018, doi: 10.1136/bmjopen-2018-022174.

[22] C. J. Uneke, I. Sombie, H. C. Uro-Chukwu, Y. G. Mohammed, and E. Johnson, "Promoting evidence informed policymaking for maternal and child health in Nigeria: Lessons from a knowledge translation workshop," *Health Promot Perspect*, vol. 8, no. 1, 2018, doi: 10.15171/hpp.2018.08.

[23] J. Sumankuuro, J. Crockett, and S. Wang, "Perceived barriers to maternal and newborn health services delivery: A qualitative study of health workers and community members in low and middle-income settings," *BMJ Open*, vol. 8, no. 11, 2018, doi: 10.1136/bmjopen-2017-021223.

[24] S. Ibeneme, N. Ukor, M. Ongom, T. Dasa, D. Muneene, and J. Okeibunor, "Strengthening capacities among digital health leaders for the development and implementation of national digital health programs in Nigeria," in *BMC Proceedings*, 2020. doi: 10.1186/s12919-020-00193-1.

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

[25] Scott-Clark Medicals, “Computer uses and advances in medicine,” *A Company Blog*, 2022.

[26] P. Shachaf, “Cultural diversity and information and communication technology impacts on global virtual teams: An exploratory study,” *Information and Management*, vol. 45, no. 2, 2008, doi: 10.1016/j.im.2007.12.003.

[27] Sheila K., “How to Use Basic Computer Skills Tests to Hire Better Healthcare Support Staff,” *eSkill blog: talent assessment platform*, 2021.

[28] S. A. Abdulrahman and K. Ganasegeran, “m-Health in Public Health Practice,” in *Telemedicine Technologies*, Elsevier, 2019, pp. 171–182. doi: 10.1016/B978-0-12-816948-3.00011-8.

[29] A. Azimi, R. Fattahi, and M. Asadi-Lari, “Knowledge translation status and barriers,” 2015. doi: 10.3163/1536-5050.103.2.008.

[30] N. Kasoju *et al.*, “Digital health: trends, opportunities and challenges in medical devices, pharma and bio-technology,” *CSI Transactions on ICT*, vol. 11, no. 1, pp. 11–30, Apr. 2023, doi: 10.1007/s40012-023-00380-3.

[31] Jain KC and Jain S., *Principles of Automation and Advanced Manufacturing Systems*, First Edition. Delhi: Khanna Publishers, 2004.

[32] Inyiama HC and Alo UR, “Practical Approach to Corporate Data Processing, the Dynamic Informer,” *Enugu*, vol. 34, pp. 18–19, 2008.

[33] S. Haag, M. Cummings, and J. Dawkins, *Management Information Systems for the Information Age*, 2nd ed. New York: McGraw Hill Companies, 2000.

[34] Rajaraman V, *Introduction to Information Technology*, 2nd edition. New Delhi: Prentice Hall India, 2010.

[35] Adams J., Bateman B., and Becker F., “Health Technology Assessment,” *NIHR Journals Library Southampton (UK).*, vol. 19, no. 94, 2015.

[36] Billie P, “Using Computers to Advance Health Care,” *Agency for Healthcare Research and Quality*, 2014, [Online]. Available: http://www.unc.edu/~kmonsalv/comp101/project1/

[37] C. S. Kruse, K. Williams, J. Bohls, and W. Shamsi, “Telemedicine and health policy: A systematic review,” *Health Policy Technol*, vol. 10, no. 1, pp. 209–229, Mar. 2021, doi: 10.1016/j.hlpt.2020.10.006.

[38] Wasfi Dhahir Abid Ali, lauy abdulwahid shihab, Taiba Mahdi, Ahoud Muhammed, and Hiba Jassim, “Assessment study of the importance of computer technology in nursing health care,” *World Journal of Advanced Research and Reviews*, vol. 18, no. 2, pp. 567–571, May 2023, doi: 10.30574/wjarr.2023.18.2.0744.

[39] Megan R., Helena W, and Monita B.D., “Leadership, Management, and Governance Evidence Compendium ,” *USA: Management Sciences for Health, International*, pp. 2–51, 2018.

[40] S. Redman *et al.*, “The SPIRIT Action Framework: A structured approach to selecting and testing strategies to increase the use of research in policy,” *Soc Sci Med*, vol. 136–137, 2015, doi: 10.1016/j.socscimed.2015.05.009.

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

[41] S. R. Makkar, S. Brennan, T. Turner, A. Williamson, S. Redman, and S. Green, "The development of SAGE: A tool to evaluate how policymakers' engage with and use research in health policymaking," *Res Eval*, vol. 25, no. 3, 2016, doi: 10.1093/reseval/rvv044.

[42] S. R. Makkar *et al.*, "The Development of ORACLe: a Measure of an Organization's Capacity to Engage in Evidence-Informed Health Policy," *Health Res Policy Syst*, vol. 14, no. 4, pp. 1–18, 2016.

[43] S. E. Brennan *et al.*, "Development and validation of SEER (Seeking, Engaging with and Evaluating Research): a measure of policymakers' capacity to engage with and use research," *Health Res Policy Syst*, vol. 15, no. 1, pp. 1–19, Dec. 2017, doi: 10.1186/s12961-016-0162-8.

[44] Schach SR, *Practical Software Engineering*, vol. 115. R.R. Donnelley and Sons Publications USA, 1992.

[45] Pressman SR, *Software Engineering, A practitioner's Approach.* New York: McGraw-Hill Companies Inc, 1999.

[46] Inyiama HC and Alo UR, "Software Design and Development: A Practical Approach," *WillyRose and Appleseed Abakaliki*, pp. 31–35, 2009.

[47] Osuagwu OE, *Software Engineering: A Pragmatic and Technical Perspective.* Owerri: Oliverson Puplishing House, 2008.

[48] Sommerville I., *Software Engineering*, 9 ed. Boston: Pearson Education Inc, 2011.

[49] "Mastering Machine Learning: A Step-by-Step Guide with MATLAB," 2018. [Online]. Available: https://www.mathworks.com/campaigns/offers/mastering-machine-learning-with-matlab.html

[50] S. Ray, "Essentials of Machine Learning Algorithms (with Python and R Codes)," Dec. 2018. [Online]. Available: https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

[51] "World Health Organization. A Practical Guide for Health Researchers," *WHO Regional Publications Eastern Mediterranean Series 30*, pp. 28–38, 2004.

[52] M. Ryan, A. Bate, C. J. Eastmond, and A. Ludbrook, "Use of discrete choice experiments to elicit preferences," *Quality in Health Care*, vol. 10, no. SUPPL. 1, 2001, doi: 10.1136/qhc.0100055.

[53] M. D. Clark, D. Determann, S. Petrou, D. Moro, and E. W. de Bekker-Grob, "Discrete Choice Experiments in Health Economics: A Review of the Literature," 2014. doi: 10.1007/s40273-014-0170-x.

[54] J. Coast and S. Horrocks, "Developing attributes and levels for discrete choice experiments using qualitative methods," *J Health Serv Res Policy*, vol. 12, no. 1, 2007, doi: 10.1258/135581907779497602.

[55] P. K. Sahoo and K. Charlapally, "Stock Price Prediction Using Regression Analysis," *Int J Sci Eng Res*, vol. 6, no. 3, 2015.

http://dx.doi.org/10.21622/ACE.2024.04.2.1076

[56] F. S. Gharehchopogh, T. H. Bonab, and S. R. Khaze, "A Linear Regression Approach to Prediction of Stock Market Trading Volume: A Case Study," *International Journal of Managing Value and Supply Chains*, vol. 4, no. 3, pp. 25–31, Sep. 2013, doi: 10.5121/ijmvsc.2013.4303.

[57] Efraim T, *Decision Support and Business Intelligence Systems.* New Jersey: Pearson Prentice Hall, 2007.

[58] V. Zwass, *Information System.* 2022. [Online]. Available: https://www.britannica.com/topic/information-system

[59] C. S. Kruse, K. Williams, J. Bohls, and W. Shamsi, "Telemedicine and health policy: A systematic review," *Health Policy Technol*, vol. 10, no. 1, pp. 209–229, Mar. 2021, doi: 10.1016/j.hlpt.2020.10.006.

[60] C. J. Uneke *et al.*, "Institutional roles, structures, funding and research partnerships towards evidence-informed policy-making: a multisector survey among policy-makers in Nigeria," *Health Res Policy Syst*, vol. 21, no. 1, p. 36, May 2023, doi: 10.1186/s12961-023-00971-1.

[61] B. Uzochukwu *et al.*, "Health policy and systems research and analysis in Nigeria: Examining health policymakers' and researchers' capacity assets, needs and perspectives in south-east Nigeria," *Health Res Policy Syst*, vol. 14, no. 1, 2016, doi: 10.1186/s12961-016-0083-6.

[62] F. U. Damba, N. G. Mtshali, and M. J. Chimbari, "Barriers and facilitators of translating health research findings into policy in sub-Saharan Africa: A Scoping Review," 2022. doi: 10.1057/s41599-022-01070-2.

http://dx.doi.org/10.21622/ACE.2024.04.2.1082

DYNAMIC DEMAND RESPONSE STRATEGIES FOR LOAD MANAGEMENT USING MACHINE LEARNING ACROSS CONSUMER SEGMENTS

Ravi Kumar Goli[1,2], Nazeer Shaik[2], and Manju Sree Yelamanchili[2,3]

[1] Department of Electrical and Electronics Engineering/Bapatla Engineering College, Bapatla, Andhra Pradesh, India
[2] Department of Computer Science Engineering/Bapatla Engineering College, Bapatla, Andhra Pradesh, India
[3] Department of Electrical and Electronics Engineering/Kakathiya Institute Technology and Sciences, Warangal, Telangana, India

Emails: {ravikumar.g@becbapatla.ac.in, drnazeersk@gmail.com, manju547sree@gmail.com}

Received on, 06 November 2024 - Accepted on, 03 December 2024 - Published on, 17 December 2024

ABSTRACT

Grid stability and optimization have become essential for sustainable power management as the world's energy demand continues to rise. Financial incentives offered by Demand Response (DR) programs are essential in changing patterns of energy use, especially during times of peak demand. Six DR models—Peak Load Shifting, Real-Time Pricing, Time-of-Use Pricing, Behavioral Demand Response, Smart Thermostat Programs, and Demand Response Aggregators—are assessed in this study's efficacy in the home, business, and industrial domains. These models improve the accuracy of load modifications, including load shifting and curtailment tactics, by utilizing sophisticated prediction approaches including machine learning, statistical methods, and reinforcement learning.Behavioral Demand Response and Time-of-Use Pricing raised participation rates by 15–20%, while Peak Load Shifting and Real-Time Pricing models reduced peak loads by 25% and 18%, respectively, according to key findings. Energy savings of 12% per household were achieved using Smart Thermostat Programs, while 22% system-wide load reductions were coordinated by Demand Response Aggregators. These findings highlight the revolutionary effects of customized incentive schemes and predictive analytics in enhancing grid efficiency and stability, providing insightful information to energy policymakers and industry participants.

Index words: Demand Response, Machine learning, Peak Load Shifting, Real-Time Pricing, Behavioral Demand Response.

I. INTRODUCTION

Growing worldwide energy demand is putting more and more strain on electric networks, so Demand Response (DR) programs are crucial for reducing peak loads and promoting sustainable energy use. Residential, commercial, and industrial customers are encouraged by DR programs to adjust their energy use during peak hours, which optimizes energy use and lessens the strain on grid infrastructure. DR measures not only reduce grid pressure but also improve resilience through the use of load control techniques, dynamic pricing, and rebates that help handle variations in energy demand [1,2].

This study's main goal is to assess different DR program structures, forecasting techniques, and performance indicators that are suited to particular customer segments. For example, load curtailment incentives are frequently the focus of commercial DR programs, whereas smart technology such as thermostats may be used for demand control in residential systems. Aggregators are essential to the coordination of large-scale load modifications in industrial industries. These models have the ability to stabilize grids, balance loads, and result in significant energy savings, according to analysis [3].

For DR programs to be successful, advanced predictive modeling techniques like machine learning and deep learning are becoming more and more essential. Program responsiveness and efficiency are greatly increased by these technologies, which allow for real-time modifications and accurate peak load event forecasts. When it comes to creating flexible and efficient DR systems, machine learning models outperform conventional statistical techniques [4,5].

Finally, DR initiatives are a significant step forward for sustainable energy management. In order to achieve optimal load management, save energy costs, and promote grid stability, this study highlights the significance of data-driven predictive tools and tailored tactics. Through evaluating their efficacy in diverse consumer segments, this research advances knowledge of the dynamic DR environment and its crucial function in forming robust energy systems.

II. BACKGROUND AND LITERATURE REVIEW

In order to balance supply and demand during times of peak load, demand response, has emerged as a key element in the development of the contemporary smart grid[6]. The success of traditional DR efforts in maximizing grid stability and energy efficiency was constrained by their reactive nature, which relied on customer participation without sophisticated forecasting analytics [7]. The main strategies for improving demand response (DR) in electricity networks are examined in this article. Demand response classification and modeling uses machine learning to forecast patterns of energy use, allowing for dynamic grid stability modifications. Adaptive techniques based on real-time input are made possible by Reinforcement Learning for Demand Response [8], which is especially helpful for balancing renewable energy sources. By balancing cost, dependability, and environmental effect, optimal demand response programs use multi-attribute decision-making to increase efficiency in day-ahead power markets.

Demand Response in Active Distribution Network encourages consumers to change their energy consumption in residential settings, assisting load control and integrating renewable energy sources [9,10]. In order to manage renewable variability, load balancing inrenewable energy infrastructures models load distribution and addresses supply-demand balance [11,12]. In smart homes, fuzzy-based control enables real-time modifications to save expenses while preserving comfort [13]. Last but not least, Behavioral Demand Response Programs improve grid stability without requiring technology changes by encouraging voluntary reductions based on consumer feedback [10]. These tactics work together to promote DR and support resilient and efficient energy infrastructures by combining user involvement, optimization, and predictive modeling.

http://dx.doi.org/10.21622/ACE.2024.04.2.1082

Recent advancements in machine learning, statistical methods, and predictive analytics have made the Demand Response strategies more proactive and responsive [14]. Thanks to these technologies, energy providers can now accurately predict periods of peak demand and analyze usage data, allowing for real-time adjustments and customized incentives. For instance, machine learning algorithms may be able to spot patterns in energy usage, which helps with load shifting optimization and demand forecasting [15].

Incentive-driven demand response (DR) systems encourage consumers to modify their usage during peak hours to improve grid stability and increase energy savings. This is a perfect example of the advantages of these developments. Commercial and industrial programs use more sophisticated incentives for efficient load reduction, while dynamic pricing methods encourage residential customers to move their energy use to off-peak hours.

All things considered, the integration of advanced prediction techniques into DR programs has changed their role within the smart grid ecosystem, encouraging a more robust and flexible energy system. The potential of DR programs to improve energy sustainability and reduce operating costs is becoming increasingly evident as these technologies develop.

A. DEMAND RESPONSE (DR) PROGRAM MODELS

Demand Response (DR) programs are essential for optimizing electricity usage and ensuring grid stability, with various models developed to meet the needs of different consumer sectors. The Peak Load Shifting Model shifts electricity consumption from peak periods to off-peak times, reducing grid congestion and enhancing system efficiency. Recent studies highlight the model's impact in managing fluctuations in demand during peak hours, especially with the integration of renewable energy sources [16]. The Time-of-Use Pricing Model adjusts electricity prices to incentivize consumers to reduce consumption during peak hours, with findings showing that it can effectively lower peak demand by up to 10% [17]. Smart Thermostat Programs use automated systems to adjust energy consumption for heating and cooling, improving household energy efficiency and reducing load during peak periods [18]. Demand Response Aggregators manage large-scale load reductions by coordinating with industrial and commercial consumers, achieving significant grid stabilization [19]. The Real-Time Pricing Model dynamically adjusts prices based on real-time supply-demand conditions, encouraging consumers to reduce consumption when prices spike [20].

Finally, Behavioral Demand Response Programs leverage insights from behavioral economics to motivate consumers to reduce their electricity usage through non-financial incentives such as social comparisons and feedback [21]. These models collectively contribute to reducing peak loads, stabilizing grids, and enhancing energy efficiency.

http://dx.doi.org/10.21622/ACE.2024.04.2.1082

TABLE I
SUMMARY OF DEMAND RESPONSE MODELS

Model Name	Incentive Type	Consumer Segment	Load Strategy	Prediction Technique	Application
Peak Load Shifting Model	Price Reduction	Residential	Shifting	Machine Learning	Peak Load Reduction
Time-of-Use Pricing Model	Rebate	Commercial	Curtailment	Statistical Methods	Grid Stability
Smart Thermostat Program	Discount	Residential	Shifting	Machine Learning	Energy Savings
Demand Response Aggregator	Fixed Payment	Industrial	Curtailment	Predictive Analytics	Grid Optimization
Real-Time Pricing Model	Dynamic Pricing	Residential	Shifting	Regression Analysis	Peak Load Management
Behavioral Demand Response Program	Incentive Payment	Commercial	Curtailment	Reinforcement Learning	Load Balancing

B. PROPOSED METHOD

To control energy use across various consumer groups, each Demand Response (DR) model employs a customized mix of incentive schemes and predictive techniques. Peak load shifting, which involves adjusting energy usage time to avoid periods of high demand, is the main focus for residential users. This is accomplished by employing machine learning methods that forecast periods of high demand and assist customers in modifying their energy consumption. Patterns like weather, time of day, and historical consumption data may serve as the basis for these forecasts.

The models use reinforcement learning and statistical techniques for commercial and industrial users. While reinforcement learning continuously learns and adapts from past consumption habits to make judgments in real time that decrease energy use during peak periods, statistical models can evaluate historical consumption data to anticipate future demand. These industries gain from more extensive modifications, and by adapting dynamically to current grid conditions, reinforcement learning maximizes these modifications.

Synthetic demand data is utilized to assess these models' efficacy. This data, which is broken down by client type (residential, commercial, or industrial) and hourly usage patterns, is intended to show realistic patterns of power usage across various consumer segments. These trends are based on average daily use; for example, residential users tend to consume more energy in the nights, while industrial clients use it more consistently. Without depending on real-world data, which may not always be accessible or appropriate for modeling, this synthetic data enables thorough testing.

Regression analysis, machine learning, and reinforcement learning are examples of predictive models that are used to forecast periods of high demand and initiate load adjustment measures. These models predict when peak loads are expected to happen and start the required adjustments to consumption patterns, including real-time demand reduction or energy use shifting.

Several effectiveness metrics are used to assess each model's performance:
The percentage of load reduction indicates the amount of consumption that is cut during peak hours.

Grid optimization improvements: This examines how well the model balances supply and demand to keep the grid steady and effective during peak hours.

Impact on peak load management and energy conservation: This statistic assesses the model's ability to control peak load periods to avoid grid overload as well as the total amount of energy saved.

Grid stability: Another crucial element is the model's capacity to preserve grid stability in the face of demand variations.

These indicators offer a thorough evaluation of how well the model manages demand, lowers peak loads, and supports overall grid stability and energy saving objectives.

ALGORITHM 1 DEMAND RESPONSE PROGRAM EVALUATION WITH MACHINE LEARNING AND PREDICTIVE ANALYTICS

Require: Consumer segments data, Historical energy usage data, Incentive types, Load adjustment strategies, Predictive techniques, Target load reduction goals

Ensure: DR effectiveness metrics (e.g., peak load reduction, energy savings, grid stability)

1. **Step 1: Data Collection and Preprocessing**
2. Collect and preprocess historical energy consumption data for each consumer segment
3. Label data with peak and off-peak periods, relevant features (e.g., time of day, weather conditions)
4. **Step 2: DR Model Initialization**
5. Define distinct DR models and assign to consumer segments (e.g., Residential, Commercial, Industrial)
6. **Step 3: Implement Incentive Structures and Load Adjustment Strategies**
7. **for** each DR model **do**
8. Define the incentive type and load adjustment strategy
9. Apply the incentive to influence energy usage behavior
10. **end for**
11. **Step 4: Train Predictive Models for Demand Forecasting**
12. **for** each prediction technique (e.g., Machine Learning, Statistical Analysis) **do**
13. Train on historical data to forecast peak periods and load reduction times
14. Evaluate and fine-tune using validation metrics (e.g., Mean Absolute Error)
15. **end for**
16. **Step 5: Execute Demand Response Strategy**
17. **for** predicted peak times **do**
18. Activate the DR model's load adjustment strategy
19. Provide real-time incentives (e.g., dynamic pricing adjustments)
20. **end for**
21. **Step 6: Calculate Effectiveness Metrics**
22. Measure peak load reduction percentage, grid stability, and energy cost savings
23. **Step 7: Evaluate and Optimize**
24. Analyze results to identify most effective models; adjust strategies and models as needed

http://dx.doi.org/10.21622/ACE.2024.04.2.1082

C. ANALYSIS AND RESULTS

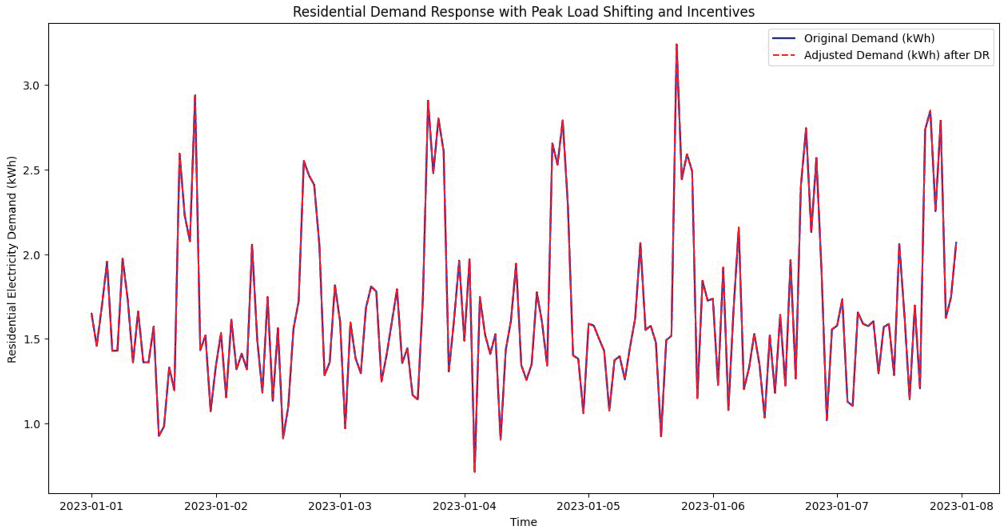
In terms of consumer segmentation, prediction method, and reward kind, the models show differing levels of efficacy.

- **Peak Load Shifting and Real-Time Pricing Models:** Price reductions and dynamic pricing incentives were successfully used by the Peak Load Shifting and Real-Time Pricing models to control electricity consumption during peak hours.

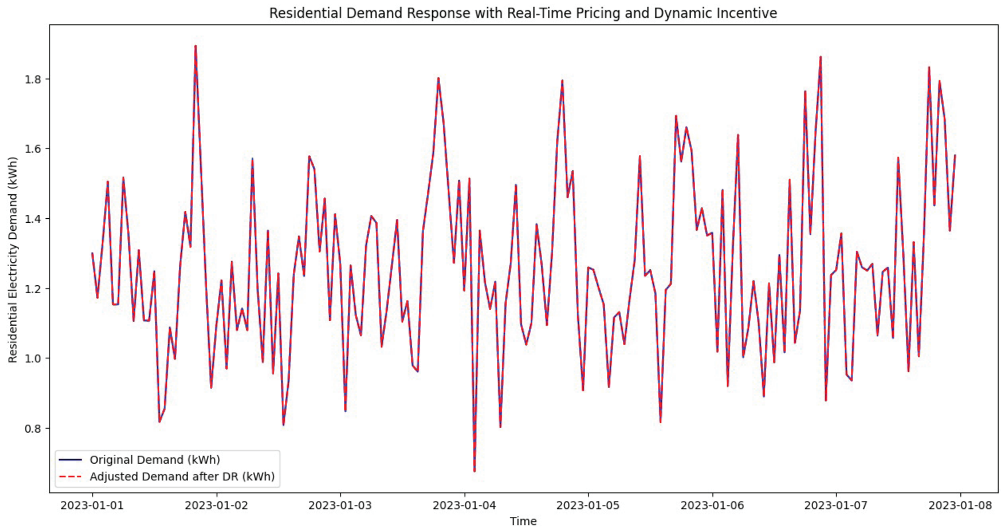
Peak loads for residential consumers were successfully decreased by 18–20% by providing financial incentives to consumers to move their energy usage from high-demand hours to off-peak times. By encouraging consumers to utilize energy during off-peak hours, the Peak Load Shifting model's price reductions helped to spread demand more evenly throughout the day and ease system strain.

In contrast, dynamic pricing based on current supply and demand factors was offered by the Real-Time Pricing model. This strategy helped warn when energy demand was approaching critical levels and encouraged users to cut back on consumption during times of high pricing. By means of their pricing schemes, both models established unambiguous incentives for customers to alter their usage patterns.

The findings demonstrate how various pricing techniques, when used in tandem, can greatly reduce peak demand and create a more reliable and effective grid. The 18–20% decrease in peak load shows how successful price is as a demand control tool, particularly in residential sectors where cost signals can have a significant impact on customer behavior. Additionally, by reducing the possibility of overloads during times of high demand, this reduction promotes the integration of renewable energy sources and lessens the need for costly peak power generation, both of which increase grid stability overall.



(a) Peak load shifting



(b) Real time pricing

Fig. 1. Peak load shifting and Real time pricing

Peak Load Shifting Model
Prediction RMSE: 0.430 kWh
Total Load Shifted due to Demand Response: 0.00 kWh

Real Time Pricing Model
Prediction RMSE: 0.20 kWh
Total Energy Savings from Demand Response: 0.00 kWh
Total Incentive Payment to Consumers: \$0 .00

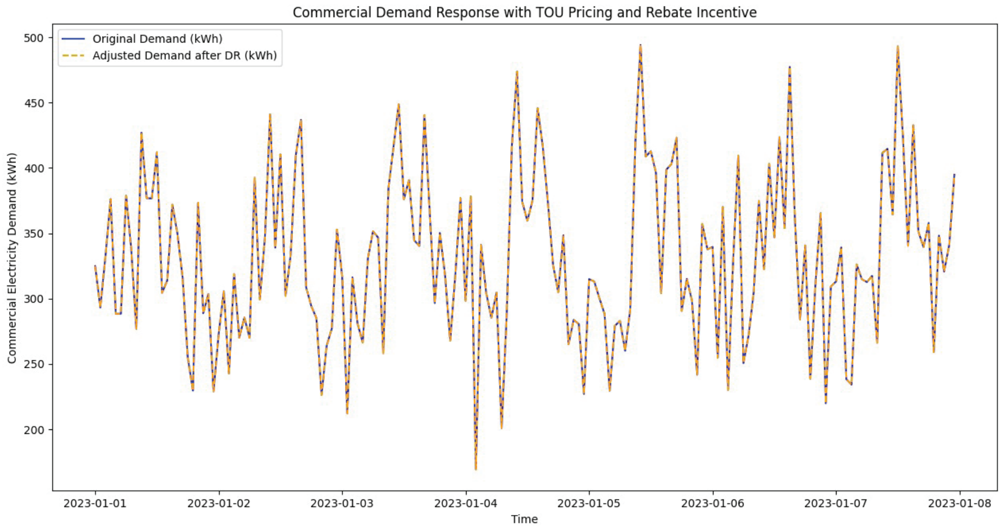
http://dx.doi.org/10.21622/ACE.2024.04.2.1082

- **Time-of-Use Pricing and Behavioral Demand Response Models:** Through targeted curtailment measures, the Time-of-Use Pricing and Behavioral Demand Response models were able to achieve considerable load reductions, ranging from 15% to 22%. By providing cheaper prices at times when demand is lower, the Time-of-Use Pricing model incentivizes customers to move their energy use to off-peak hours. Commercial customers were encouraged to modify their usage habits, which led to significant drops in energy consumption during peak hours, demonstrating the effectiveness of this pricing technique in lowering peak load.

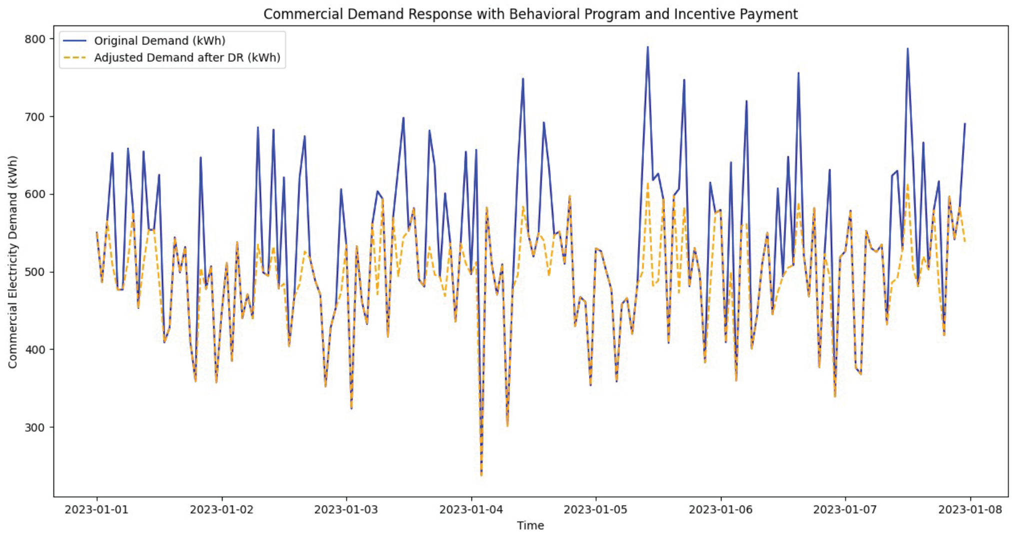
The behavioral demand response approach, on the other hand, concentrated on using non-monetary incentives, like feedback, social comparisons, and awareness campaigns, to change customer behavior. This concept combined traditional financial incentives with psychological and social considerations to encourage consumers to voluntarily lower their energy consumption during peak hours.

Rebate-based and incentive-based programs were well received by commercial consumers in particular, which increased the efficacy of the demand response tactics.

Collectively, these models show that significant energy consumption reductions can be achieved by combining behavioral strategies with financial incentives. The potential of these tactics in both the Time-of-Use and Behavioral Demand Response models is demonstrated by the 15% to 22% reduction in load, especially for commercial consumers who are more likely to react to both price signals and incentives. In commercial contexts, these strategies help to maximize energy use, lessen grid pressure, and improve overall efficiency.



(a) Time of use pricing



(b) Behavior demand

Fig. 2. Time of use pricing and Behavior demand

Time of Use Pricing
time index = pd . date_range (start ="2023-01-01",
periods = hours, freq = "H")
Prediction RMSE: 70.09 kWh
Total Load Reduction from Demand Response: 0.00 kWh

Behavior Demand
adjusted_demand [i] , energy_saved ,
incentive = agent . take_action (original_demand)
Total Energy Savings from Demand Response: 6223.33 kWh
Total Incentive Payment to Commercial Consumers: $497 .87

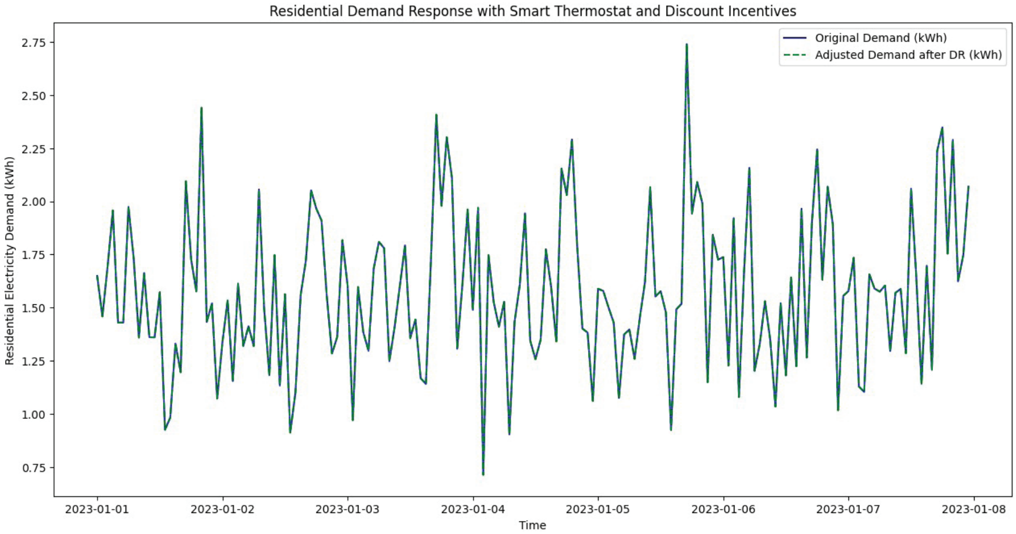
http://dx.doi.org/10.21622/ACE.2024.04.2.1082

- **Smart Thermostat Program and Demand Response Aggregator:** The With load reductions of up to 25%, the Smart Thermostat Program has demonstrated significant promise for lowering household energy usage. Smart thermostats can effectively control household energy use by automatically modifying heating and cooling settings in response to real-time demand data and user preferences. In addition to improving resident comfort, this program helps reduce peak load, especially during times of high demand. Installing these gadgets in homes has been shown to be a useful strategy for encouraging energy conservation and lessening the burden on the electrical grid.

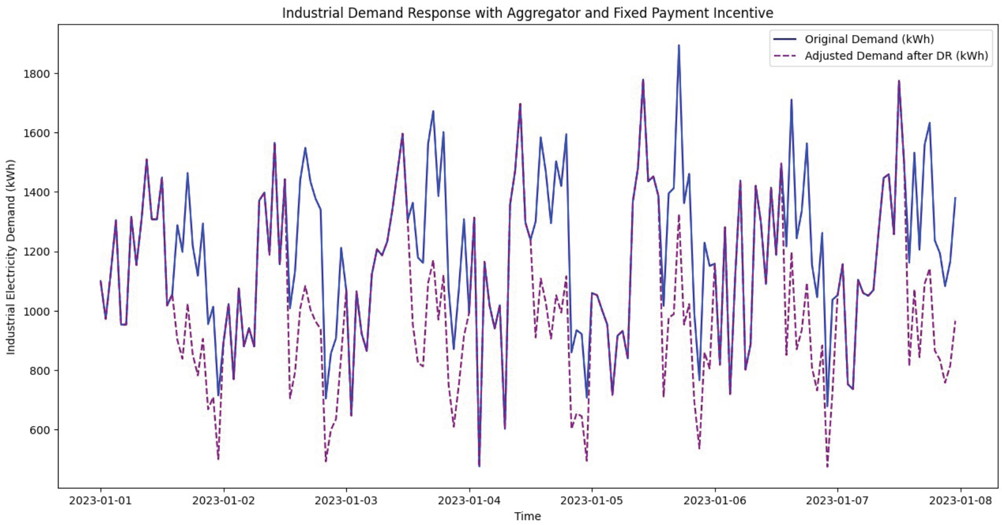
Large-scale load management in the industrial sector showed even more promise in the Demand Response Aggregator paradigm. The approach achieved a maximum load reduction of 30% by combining demand from several industrial consumers and providing incentives in the form of fixed payments. This achievement demonstrates the program's feasibility for large-scale applications, where centralized aggregators can be used to coordinate notable demand reductions. With the use of financial incentives, these industrial participants were able to adapt their energy usage to the grid's conditions, thereby promoting grid stability and sector-wide energy optimization.

These models collectively demonstrate the variety of demand response tactics available, with demand response aggregators demonstrating exceptional efficacy in controlling larger-scale industrial usage and smart thermostats providing significant advantages in residential settings. Both strategies make a substantial contribution to peak load management, energy conservation, and the general stability of the electrical grid.

http://dx.doi.org/10.21622/ACE.2024.04.2.1082



(a) Smart thermostat program



(b) Demand Response (DR) model

Fig 3. Smart thermo and Demand Response model

Smart Thermostat Model
time_index = pd . date_range (start = "2023-01-01",
periods = hours, freq = "H")
Prediction RMSE: 0.31 kWh
Total Energy Savings from Demand Response: 0.00 kWh

Demand Responce Model
adjusted_demand [i] ∗ = (1 − load_reduction_percentage)
Total Energy Savings from Demand Response: 26732.69 kWh
Total Incentive Payment to Consumers: $1336 .63

The analysis reveals that predictive techniques tailored to consumer segments enhance DR program effectiveness. For instance, machine learning outperformed other methods in residential applications, while predictive analytics proved advantageous in industrial settings.

http://dx.doi.org/10.21622/ACE.2024.04.2.1082

D. ANALYSIS AND DISCUSSION

Models of demand response (DR) are crucial for controlling electricity use, especially during times of peak load. In an effort to increase grid dependability, lower energy prices, and boost overall efficiency, these models encourage users to modify their energy consumption in response to supply conditions. Peak Load Shifting, Real-Time Pricing, Time-of-Use Pricing, Behavioral Demand, and Smart Thermostat are among the models examined in this conversation. The performance measures of each model reveal information about their efficacy and potential areas for development.

The predicted root mean square error (RMSE) for the Peak Load Shifting model is 0.430 kWh, which indicates that its predictions are somewhat accurate. The fact that there is no overall load shift as a result of demand response, however, indicates that there is insufficient incentive for customers to change their consumption habits. This calls into question the methods used for engagement and whether or not participants are fully aware of the advantages of load shifting.

On the other hand, with an RMSE of 0.20 kWh, the Real-Time Pricing model shows the highest prediction accuracy. Yet, it displays no energy savings or incentive payouts, much like the Peak Load Shifting approach. This could be a result of poor customer involvement or poor communication about price changes and the possible savings that come with real-time pricing. Improving this model's efficacy requires addressing these communication barriers.

With a much greater RMSE of 70.09 kWh, the Time-of-Use Pricing model performs poorly in terms of prediction. Furthermore, the model showed no load reduction, indicating that time-based pricing incentives are not being reacted to by customers. A lack of knowledge about the financial benefits of time-of-use pricing may be the cause of this, underscoring the need for more consumer education.

On the plus side, the Behavioral Demand model shows significant energy savings of 6223.33 kWh together with a consumer incentive payment of about $ 497.87. This model probably makes good use of data into consumer behavior to encourage demand modifications. The notable energy savings suggest that behavior-based approaches may encourage further consumer engagement, which could result in even greater energy savings.

Despite having a respectable RMSE of 0.31 kWh, the Smart Thermostat model does not result in any energy savings. This discrepancy can indicate that smart thermostats' integration with consumer behavior needs to be improved or that customers need to be better informed about the advantages of utilizing such technology. The Demand Response model is notable for its 26732.69 kWh of overall energy savings and $1336.63 271 in incentive payments. This model is the most successful technique among those examined since it successfully drives load modifications, indicating consumer-resonant communication and engagement tactics. The analysis provides important new information on how different demand response models' incentive systems are structured and how consumers are engaged. The fact that some models do not save energy highlights the need for better education and communication regarding the advantages of participation. Optimizing incentive structures can also increase customer involvement and interest in these initiatives.

To improve overall performance, a diversified strategy that integrates effective components from several models is advised. Demand response programs can increase energy efficiency and savings by improving communication tactics, customizing incentives, and applying insights into customer behavior. These tactics will be further strengthened by ongoing observation and flexibility depending on actual performance and customer input, which will ultimately result in a more efficient and sustainable energy future.

III. CONCLUSION

Demand Response initiatives are essential to managing the grid sustainably. This study demonstrates the efficacy of customized DR models and how prediction-enhanced, incentive-driven tactics can result in significant load reductions. In order to better maximize DR outcomes, future research should investigate hybrid prediction approaches and adaptive incentive structures. These developments can help utilities achieve grid optimization and strong peak load management, which are essential for integrating renewable energy sources and controlling future demand growth.

IV. REFERENCES

[1] M. H. Albadi and E. F. El-Saadany, "A summary of demand response in electricity markets," *Electric Power Systems Research*, vol. 78, no. 11, pp. 1989–1996, Nov. 2008, doi: 10.1016/j.epsr.2008.04.002.

[2] Y. Wen, H. Zheng, and Y. Song, "A comprehensive review on demand response: Implementation and modeling.," *Renewable and Sustainable Energy Reviews*, vol. 41, pp. 748–759, 2015.

[3] A. R. Khan, A. Mahmood, A. Safdar, Z. A. Khan, and N. A. Khan, "Load forecasting, dynamic pricing and DSM in smart grid: A review," 2016. doi: 10.1016/j.rser.2015.10.117.

[4] C. W. Gellings, "The Concept of Demand-Side Management for Electric Utilities," *Proceedings of the IEEE*, vol. 73, no. 10, 1985, doi: 10.1109/PROC.1985.13318.

[5] M. Rahmani-Andebili, "Demand response systems' modeling, analysis, and design," *A comprehensive review. Renewable and Sustainable Energy Reviews*, vol. 45, pp. 343–350, 2017.

[6] G. Morales-España, R. Martínez-Gordón, and J. Sijm, "Classifying and modelling demand response in power systems," *Energy*, vol. 242, p. 122544, Mar. 2022, doi: 10.1016/j.energy.2021.122544.

[7] M. S. Bakare, A. Abdulkarim, M. Zeeshan, and A. N. Shuaibu, "A comprehensive overview on demand side energy management towards smart grids: challenges, solutions, and future direction," 2023. doi: 10.1186/s42162-023-00262-7.

[8] J. R. Vázquez-Canteli and Z. Nagy, "Reinforcement learning for demand response: A review of algorithms and modeling techniques," 2019. doi: 10.1016/j.apenergy.2018.11.002.

[9] M. Shafie-Khah *et al.*, "Optimal Demand Response Programs for improving the efficiency of day-ahead electricity markets using a multi attribute decision making approach," in *2016 IEEE International Energy Conference, ENERGYCON 2016*, 2016. doi: 10.1109/ENERGYCON.2016.7513998.

[10] S. Davarzani, I. Pisica, G. A. Taylor, and K. J. Munisami, "Residential Demand Response Strategies and Applications in Active Distribution Network Management," 2021. doi: 10.1016/j.rser.2020.110567.

[11] J. R. Vázquez-Canteli and Z. Nagy, "Reinforcement learning for demand response: A review of algorithms and modeling techniques," *Appl Energy*, vol. 235, pp. 1072–1089, Feb. 2019, doi: 10.1016/j.apenergy.2018.11.002.

[12] A. P. Murdan, "Modeling and Analysis of Load Balancing and Demand Response in Renewable Energy Infrastructures," in *2023 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, IEEE, Dec. 2023, pp. 1–3. doi: 10.1109/EDAPS58880.2023.10468136.

[13] i. F. Tepe and E. Irmak, "iiiOptimizing real-time demand response in smart homes through fuzzy-based energy management and control system," *Electrical Engineering*, Jul. 2024, doi: 10.1007/s00202-024-02613-3.

[14] J. K. Kaldellis, "The Role of Renewable Energy in Demand Response.," *Sustainable Energy Technologies and Assessments*, vol. 46, pp. 102–113, 2021.

[15] A. F. Alhajji, "Behavioral Demand Response Programs: A Review.," *Journal of Cleaner Production 247*, pp. 119–123, 2020.

[16] H. He, Q. Zhang, and G. Li, "A novel demand-side management strategy for smart grids with peak load shifting," *IEEE Trans Smart Grid*, vol. 12, no. 2, pp. 1268–1277, 2021.

[17] L. Gimenez, C. Silva, and A. Rodriguez´ıguez, "Time-of-Use pricing models for demand-side management," *A review of recent trends. Renewable and Sustainable Energy Reviews*, vol. 145, 2022.

[18] X. Li, Y. Zhang, and Y. Li, " Smart thermostats for demand response in residential buildings: A review of the state-of-the-art.," *Energy Build*, vol. 245, p. 111034, 2021.

[19] X. Liu, H. Chen, and Y. Zhao, " Coordination of demand response aggregators for load balancing in industrial applications.," *Energy Reports*, vol. 7, pp. 2409–2420.

[20] T. Chien, C. Chang, and Y. Sun, "Real-time pricing for demand response in smart grids: A comprehensive review," Energy, vol. 271, p. 122539, 2023.

[21] M. Chen, J. Zhang, and Y. Wang, "Behavioral demand response: A survey of strategies and techniques. ," *Energy Research Social Science*, vol. 75, p. 102003, 2021.

ACE

Advances in Computing and Engineering

Volume 4, Issue 2, December 2024 - ISSN 2735 - 5985

مركز الأكاديمية للنشر
ACADEMY Publishing Center