Utilizing the Bees Algorithm and Machine Learning to optimize the UAVs travel time in consideration of Weather Conditions

Vedant Zope[1], Asrul Harun Ismail[2]

[1] Indian Institute of Technology Kharagpur,
Email : vedantzope@kgpian.iitkgp.ac.in
[2]College of Engineering and Physical Sciences, The University of Birmingham
Email : asrul.ismail@port.ac.uk

Abstract. Unmanned Aerial Vehicles (UAV) speed is sensitive to wind disturbances and due to the increasing demand for the use of UAVs in outdoor environments, it is essential to pay special attention to all of the factors that influence the deployment of UAVs for various reasons. Among these are UAV specification (including battery capacity), and energy consumption (heavily influenced by weather conditions and payload). All of these factors must be taken into account for the optimum deployment of UAVs. This work develops a model for calculating optimal routing plans in the presence of wind disturbances using the Bees Algorithm and Machine Learning.

Keywords: Bees Algorithm, Combinatorial optimization Problem(COP), Metaheuristics Algorithm, Asymmetric Travelling Salesman Problem.

1 Introduction

Unmanned Aerial Vehicles (UAVs), often referred to as drones, have garnered significant attention in recent years due to their versatility and wide range of applications. From commercial delivery services [1] to military operations [2] and even healthcare [3], the potential applications of UAVs are vast and continuously expanding. However, as these applications grow, so do the challenges associated with ensuring efficient and safe UAV operations.

One of the primary challenges in UAV operations is routing optimization. Traditional routing optimization research focuses on optimizing the distance, aiming to find the shortest path between points. While this is a crucial aspect, it often overlooks the dynamic nature of real-world scenarios where external factors, especially weather conditions, play a pivotal role. Weather conditions, such as wind speed and direction, can drastically impact the performance and travel time of UAVs [4]. As such, merely optimizing for distance can lead to suboptimal routes in real-world applications.

Existing solutions in UAV routing have predominantly centered around combinatorial optimization techniques [5]. While these methods offer robust solutions for distance optimization, they often fall short when introduced to dynamic real-world challenges like unpredictable weather patterns.

In this study, we introduce a novel approach to UAV routing optimization: the Combinatorial Bees Algorithm with Machine Learning (CBA-ML). Our approach not only considers the fundamental aspect of distance optimization but also integrates insights from machine learning to account for the effects of weather on UAV velocity. By doing so, we aim to provide a more holistic, practical, and efficient solution to UAV routing, particularly pertinent to real-life scenarios such as delivery services, military operations, and emergency medical responses.

2 Background

2.1 *Travelling Salesman Problem(TSP)*

TSP is an NP-hard combinatorial optimization problem that is concerned with determining the best routing between a central depot and several nodes scattered in the space, the distance between two cities is the same in each opposite direction, forming an undirected graph. This symmetry halves the number of possible solutions. In asymmetric TSP, paths may not exist in both directions or the distances might be different, forming a directed graph. Traffic collisions, one-way streets, and airfares for cities with different departure and arrival fees exemplify how this symmetry could break down symmetric TSP undirected graph asymmetric TSP. Optimizing UAVs travel time while considering weather conditions would classify as an asymmetric TSP, as the weather would influence the travel time. Hence, the time taken for the UAV to get from the city i to the city j would not necessarily be the same as that of the city j to the city i.

2.2 *Bees Algorithm for Combinatorial Optimization Problems*

This section will explain the main element of BA and how it is applied to the VRP. The basic BA, which was initially developed in 2005 to solve continuous problems [6], has been improved to its combinatorial versions on solving PCB assembly, timetabling, machine scheduling, and TSP. Combinatorial and continuous versions of BA principally have a similar general procedure but completely different search principles. The combinatorial domain has the following main characteristics: the search space is discrete; the constraints are finite; the solution has an ordered sequence and a cost function related to the combination. Since the BA was originally developed to solve continuous domains, changing the searching section in global and local search with a discrete search operator is essential. Hence in the combinatorial version, a discrete random generator is used, replacing the real number generator in the continuous version.

3 Related Work

In the evolving field of UAV routing optimization, numerous algorithms and methodologies have been established. Traditional techniques predominantly concentrate on minimizing the travel distance of UAVs. For instance, the Particle Swarm Optimization (PSO) technique has been employed in various UAV optimization tasks, notably for path planning in obstacle-rich environments [7]. Genetic Algorithms, another evolutionary technique, have been employed in path planning endeavors, showcasing their potential in solving complex combinatorial problems pertinent to UAV routing [8].

Real-world UAV operations are often subjected to various challenges, with weather being a non-trivial factor. A study by R. Clothier et al. [9] explores the risk assessment of UAV operations, considering various real-world factors, including weather conditions.

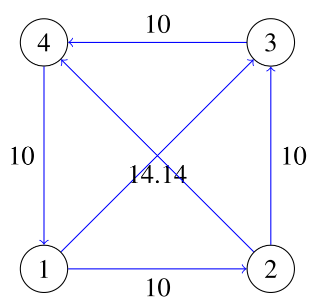
The Combinatorial Bees Algorithm (CBA) [10], while primarily researched in different optimization contexts, has its potential yet to be fully realized in the UAV routing domain. Its success in other areas, such as job scheduling problems, underscores its potential applicability in UAV routing, especially when real-life challenges, like weather effects, are factored in.

Our work uniquely integrates the nuances of real-world challenges(taking into account actual weather conditions) with the combinatorial optimization potential of the Bees Algorithm, augmented by machine learning insights to offer a practical, efficient UAV routing solution.

4 Problem Formulation

Optimizing a UAV's route is fundamentally similar to the well-known Travelling Salesman Problem (TSP), where the goal is to find the shortest path through a set of points. However, for UAVs, wind can change flight times depending on the direction. This makes our problem a type of TSP where travel between two points isn't always equal in both directions, known as an asymmetric TSP (ATSP)

4.1 Case with No Wind



Velocity Matrix:

	1	2	3	4
1	0	10	10	10
2	10	0	10	10
3	10	10	0	10
4	10	10	10	0

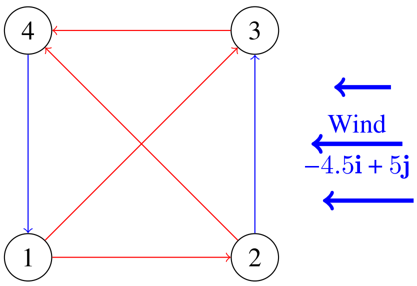
Duration Matrix:

	1	2	3	4
1	0	1	1.414	1
2	1	0	1	1.414
3	1.414	1	0	1
4	1	1.414	1	0

Route Duration:

Route	1-2	2-3	3-4	4-1
Distance	10	10	10	10
Duration	1	1	1	1

4.2 Case with External Wind



Velocity Matrix (considering Wind Effect):

	1	2	3	4
1	0	5.5	16.06	15.34
2	14.5	0	7.42	23.53
3	15.34	23.53	0	5.5
4	16.06	7.42	14.5	0

Distance Matrix:

	1	2	3	4
1	0	10	14.14	10
2	10	0	10	14.14
3	14.14	10	0	10
4	10	14.14	10	0

Route Duration (Direction 1):

Route	1-3	3-4	4-2	2-1
Distance	14.14	10	14.14	10
Duration	0.88	1.82	0.6	0.69

Route Duration (Direction 2):

Route	1-2	2-3	3-4	4-1
Distance	10	10	10	10
Duration	1.82	1.35	1.82	0.65

In the real-world UAV path planning, the influence of wind can't be overlooked. Wind affects the actual velocity of UAVs, which in turn impacts the time taken to move between nodes. A UAV navigating against the wind would generally take longer to reach its destination compared to a UAV navigating with the wind or perpendicular to it. The matrices above illustrate how the wind with components $-4.5\mathbf{i} + 5\mathbf{j}$ affects the UAV's velocity and subsequently, its travel durations between nodes. This factor necessitates an adaptive path planning method, where routes are determined not only based on distances but also on prevailing wind conditions. This element of variability introduced by the wind establishes our problem's unique and dynamic nature in the realm of UAV path planning research.

5 Mathematical Model

Asymmetric TSP's graph is represented by a directed weighted graph G=(V,E), where V is a set of vertices representing the cities, and all the connecting lines between the cities are E. Every edge indicates a possible route between two connected vertices or cities[11].

The variable Ti,j is associated with an edge (i,j) and represents the time duration taken to travel the Euclidean distance from vertex $(x_i, y_i)\ to\ (x_j, y_j)$ as Equation (1). Before executing the BA, the time taken by all edges has been calculated and stored as a duration matrix(T) as shown in Equation(2).

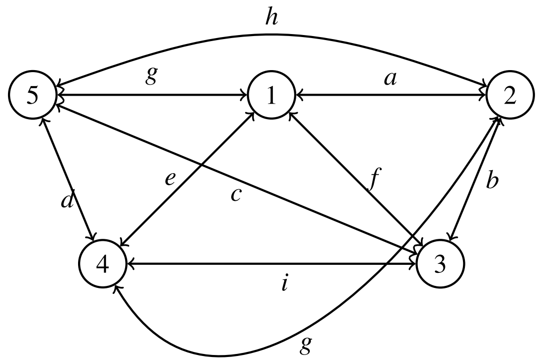


Fig. 1: Visual representation of the duration matrix T

$$T = \begin{bmatrix} 0' & a & f & e & g \\ a' & 0 & b & g & h \\ f' & b' & 0 & i & c \\ e' & g' & i' & 0 & d \\ g' & h' & c' & d' & 0 \end{bmatrix} \qquad T_{i,j} = \frac{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{v_{i,j}} \tag{1}$$

$$if\ i < j : time = x$$

$$elseif\ i > j : time = x'$$

The objective is to find the minimal total tour time of the final closed Hamilton cycle (only visiting once of all cities) tour as defined in Equation (3).

$$\sum_{n=1}^{n-1} T_{i,i+1} + T_{n,1} \tag{2}$$

6 Methodology (CBA — ML)

6.1 Machine Learning Approach for UAV Speed Prediction

In the intricate dance between UAV parameters and wind dynamics, the underlying patterns governing the final UAV speed require sophisticated methods to decode. To achieve this, our methodology incorporates a robust machine-learning framework.

Firstly, an **exploratory data analysis (EDA)** was conducted. This is a critical stage in our methodology, as it allows for an initial assessment of potential patterns, correlations, or anomalies in the dataset. It guided our preprocessing steps and provided insights into which machine learning models might be most appropriate for this task.

Following EDA, we proceeded with **data preprocessing**. Given the varied nature of the data and its sources, preprocessing ensures that the dataset is primed for model training—removing any inconsistencies, handling missing values, and ensuring that categorical variables are correctly encoded.

Central to our approach is the **model selection** stage. Instead of relying on a single model, multiple machine learning algorithms were evaluated. The choice of the best model is based on the Mean Squared Error (MSE) score, which provides an objective measure of how well each model can predict the UAV speed. After rigorous evaluation, the CatBoost algorithm—a gradient boosting method on decision trees—emerged as the most promising. Its inherent ability to handle categorical features directly, combined with robust performance, made it a clear choice.

For achieving peak performance, **hyperparameter tuning** was conducted using Ray Tune. This fine-tuned the CatBoost model to ensure maximum accuracy in predicting the UAV speed, which is essential for the subsequent integration with the Combinatorial Bees Algorithm.

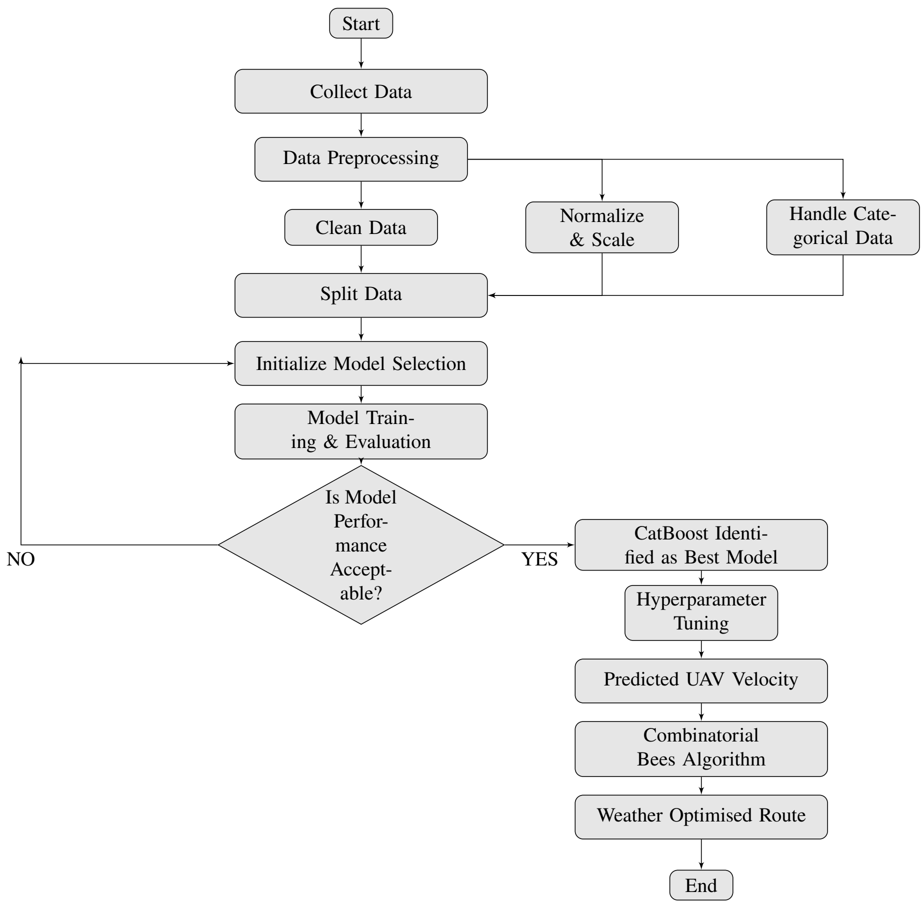


Fig. 2: Flowchart of the proposed method integrating CatBoost with Combinatorial Bees Algorithm for UAV velocity prediction and routing optimization(CBA-ML)

6.2 Data Utilized in this Study

A note on the data utilized: in this study, we worked with **dummy data**. This data was meticulously crafted using historical weather datasets sourced from the OpenWeatherMap API. It simulates the potential influences of various weather conditions on UAV speed and its parameters, providing a realistic yet hypothetical scenario for our machine-learning framework.

6.3 Combinatorial Bees Algorithm

Before beginning with BA, we need to set the parameters initially [scout bees *(n)*, elite bees *(nep)*, best bees *(nsp)*, elite sites *(e)*, and best sites *(m)*], all the parameters are set dependently on *(n)*.

No. of elite bees (nep) equal to No. of scout bees (n), No. of best bees (nsp) equal to 50% of No. of elite bees (nep), No. of elite sites (e) equal to 50% of No. of best sites (m), No. of best sites (m) equal to 50% of No. of scout bees (n).

The generation of the initial solution begins with a population of scout bees, denoted as n. These are constructed using random permutation. Following this construction, promising solutions from this initial batch are further exploited. Both elite sites, represented as e, and the best sites, represented as m, are areas designated for more detailed exploration by the worker bees. These bees, quantified as nep for those examining elite sites and nsp for those examining best sites, utilize neighborhood or local search mechanisms. Specifically, the neighborhood search operator implements a basic form of local search. This method combines swap, insertion, and reversion operations, each with an equal probability of 33% when applied in the context of the combinatorial bees algorithm. The remaining bees, which can be represented by the equation $n - e - m$, will embark on a journey to explore other solution spaces, primarily utilizing the global search operator.

Table 1: Parameter setting on BA

Parameter	Value
Number of scout bees (n)	n
Number of elite bees (nep)	n
Number of best bees (nsp)	0.5n
Number of elite sites (e)	0.25n
Number of best sites (m)	0.5n

Table 2: The parameters of BA are set as shown above and are taken from the previous study of the basic Bees Algorithm for the combinatorial domain by Ismail et al. [10]

Along with other initial parameters, the number of iterations also needs to be fixed, the number of iterations is to be set in such a way that the solution converges close to the optimal solution, and at the same time, the number of iterations is as small as possible such that the computational time required is less.

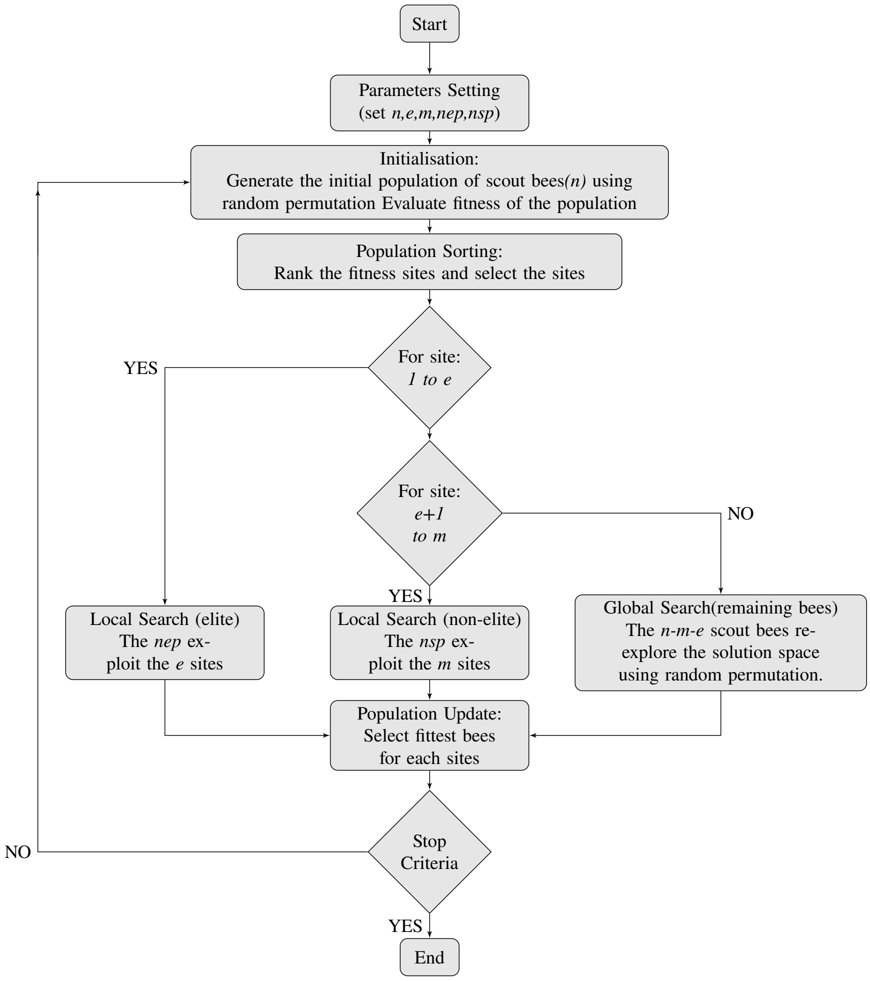


Fig. 3: Flowchart Explaning working of Combinatorial Bees Algorithm

6.4 Analysis of Iteration Convergence

The iterative nature of optimization processes often leads to a pivotal question: how many iterations are sufficient to achieve convergence? In the context of our analysis, the convergence of the best cost is closely tied to the number of nodes in the Hamiltonian cycle. Specifically, a noticeable trend emerged: the number of iterations necessary for convergence approximated the number of customer nodes thrice. Formally:

$$\text{Number of Iterations} \approx 3 \times \text{Nodes}$$

Figure 4 visually represents this relationship. As the number of nodes grows, there's a near-linear increase in the required iterations, effectively capturing the essence of the

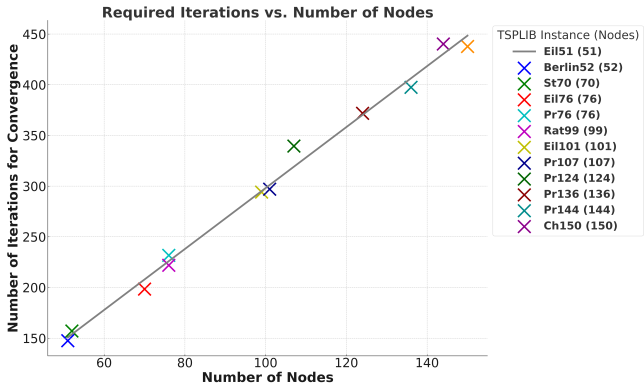


Fig. 4: Relationship between the number of nodes and the required iterations for convergence. The gret line represents a linear fit, reinforcing the observed trend.

$3 \times$ Nodes observation. This behavior is underpinned by the grey linear fit depicted in the plot, the equation of which mirrors our initial hypothesis. This relationship can be particularly useful when estimating computational efforts for larger node sets, thus optimizing the resources utilized during the optimization process.

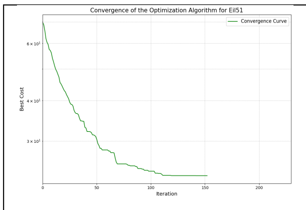
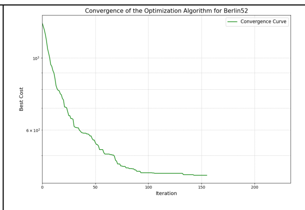
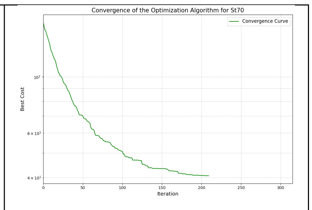
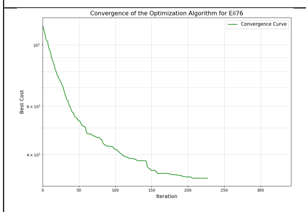
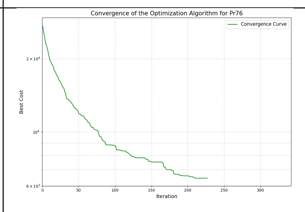
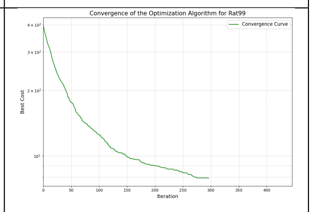
		
Eil51	Berlin52	St70
		
Eil76	Pr76	Rat99

Table 3: Cost vs Iteration plots for the studies.

Through these visual representations, the studies display a consistent trend in cost reduction as iterations progress, which reinforces the algorithm's capability to hone in on an optimal or near-optimal solution. Moreover, by comparing the curves, one can deduce the relative difficulty or complexity of each dataset in terms of reaching the optimal solution.

7 Results and Discussion

The core objective of our study was to develop an algorithm that integrates the impact of weather into routing optimization. Traditional Traveling Salesman Problem (TSP) algorithms prioritize distance minimization. While effective in many scenarios, such strategies might be suboptimal for applications involving Unmanned Aerial Vehicles (UAVs). In UAV routing, weather conditions, particularly wind, can significantly influence the route's efficiency and safety.

CBA-ML offers an innovative amalgamation of the Combinatorial Bees Algorithm with Machine Learning. While the Combinatorial Bees Algorithm shoulders the responsibility of routing, Machine Learning adeptly handles the dynamic assessment of wind and other atmospheric interferences. This synergy allows CBA-ML to deliver optimized routes that reflect real-world conditions.

Our experimental setup employed six benchmark datasets from TSPLIB. The parameters were: 40 scout bees (n), 40 elite bees (nep), 20 best bees (nsp), 10 elite sites (e), 20 best sites (m), and 3× Nodes iterations.

Table 4: Results of CBA-ML

Datasets	**Best CBA (Time)**	**Best CBA-ML (Time)**	**% Time Reduction**
Eil51	45.38	23.85	47.42%
Berlin52	904.8	482.71	46.68%
St70	72.8	40.35	44.55%
Eil76	62.9	32.75	47.99%
Pr76	10918.9	6699.17	38.63%
Rat99	173.9	91.43	47.38%

Table 4 offers a comprehensive view of our experimental outcomes. We took CBA as our foundational reference, which solely prioritizes route optimization without the nuanced consideration of wind dynamics. In contrast, CBA-ML, with its integrated approach, brings in the wind dynamics, providing a richer, more realistic optimization.

The empirical evidence, as presented, underscores the edge CBA-ML holds over the traditional CBA. The edge is not just in raw efficiency but in its ability to make nuanced decisions. Table **??** visually contrasts the routes optimized by both algorithms. Intriguingly, CBA-ML, informed by the Machine Learning model, often leans towards paths that might be longer in distance but prove to be more efficient when weather variables come into play.

It's important to understand the underlying philosophy of such choices: it's not about the shortest distance, but the fastest and safest route. Wind resistance or tailwinds can dramatically influence UAV energy consumption and flight stability. Thus, a slightly longer route with favorable winds might be more efficient than a shorter one against strong winds. Our results, showcasing efficiency improvements from 38.63% to an impressive 47.99%, validate this approach. Moreover, larger urban datasets underscore the real-world applicability and relevance of CBA-ML, positioning it as a potential game-changer in UAV routing, especially in densely populated urban settings.

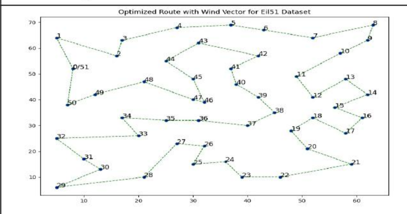
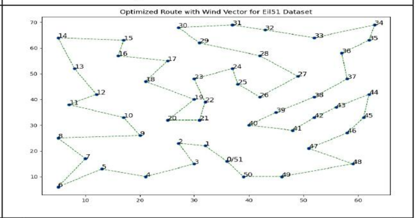
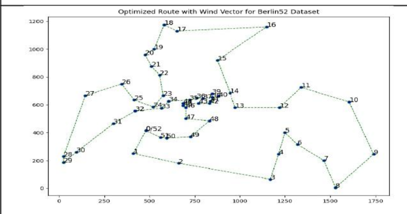
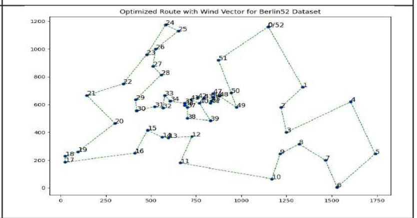
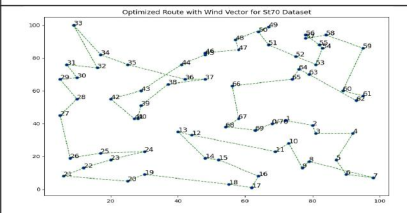
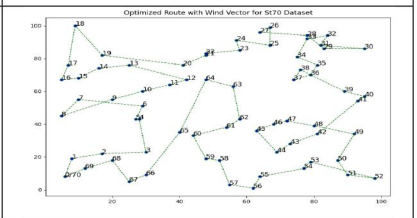
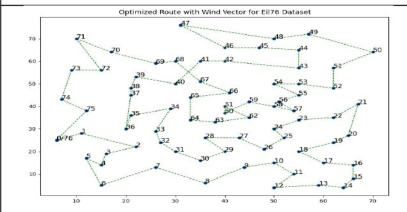
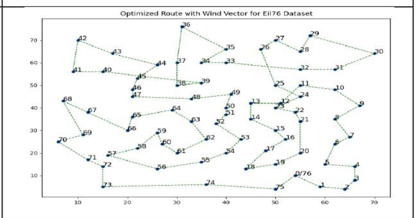
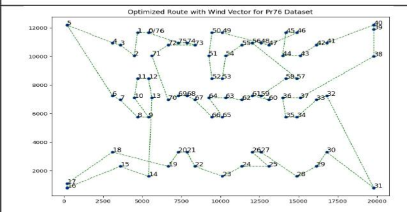
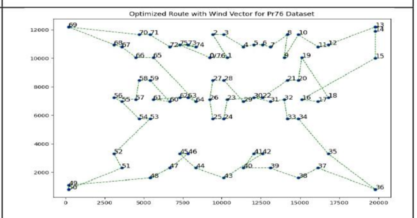
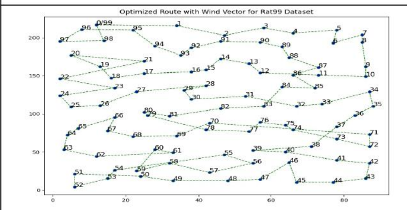
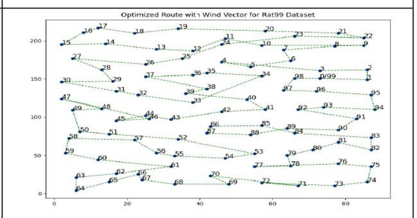
Dataset	CBA Optimized Route	CBA-ML Optimized Route
Eil51		
Berlin52		
St70		
Eil76		
Pr76		
Rat99		

Table 5: Visual comparison of optimized routes using CBA and CBA-ML for each study.

8 Conclusion

In this research, we presented an augmented version of the Combinatorial Bees Algorithm (CBA) by seamlessly integrating machine learning, leading to the development of CBA-ML. The essence of our approach was to bridge the gap between traditional combinatorial optimization techniques and the dynamic factors affecting UAV routing.

Our experimentation, conducted using six datasets from the renowned TSPLIB, supplied compelling empirical evidence. Notably, it's essential to clarify that some of our tests employed dummy data, acting as a surrogate for real-world scenarios. While this dummy data offered valuable insights and allowed for methodological development and testing, it's imperative to transition to actual real-world data in subsequent studies. Such a shift from simulated to tangible data is paramount to refine our approach in diverse and intricate scenarios.

Furthermore, the results showcased a significant enhancement in routing efficiency when CBA-ML was juxtaposed with traditional CBA. This improvement accentuates the necessity of embedding dynamic factors, such as wind patterns, into routing algorithms, especially considering real-world applications for Unmanned Aerial Vehicles (UAVs).

It's also worth mentioning that our research primarily concentrated on routing optimization considering weather patterns and didn't delve into the realms of path planning and obstacle avoidance. These areas, though outside the purview of our current study, represent crucial facets of UAV navigation and present potential avenues for future research integration.

While our current foray focused on a particular subset of potential scenarios, a broader horizon beckons. Future research should consider a larger set of cities, implications of deploying multiple UAVs, and introducing time windows and other real-world constraints. Such endeavors will truly assess CBA-ML's potency for genuine UAV mission planning and deployment.

In wrapping up, the merger of machine learning with combinatorial algorithms, as exhibited in our study, holds immense promise in the domain of UAV routing. However, a vast spectrum of real-world challenges and integrations, such as path planning and obstacle avoidance, remain to be explored, urging continued research and enhancement of the introduced methodologies.

Data and Code Availability

For the sake of reproducibility and further exploration, the code and datasets used in this study have been made available on GitHub. The repository can be accessed at https://github.com/vedantzope/UAV-Weather-Optimized-Routing.

Bibliography

[1] A. Goodchild and J. Toy, "Delivery by drone: an evaluation of unmanned aerial vehicle technology in reducing co2 emissions in the delivery service industry," *Transportation Research Part D: Transport and Environment*, vol. 61, pp. 58–67, 2017.

[2] I. Maza, F. Caballero, J. Capitán, J. R. Martínez-de Dios, and A. Ollero, "Multiple uav cooperative searching operation using polygon area decomposition and efficient coverage algorithms," in *Distributed Autonomous Robotic Systems*. Springer, 2011, pp. 221–230.

[3] J. Rosen, "Drones and the future of disaster response," *New England Journal of Medicine*, vol. 376, no. 24, pp. 2305–2307, 2017.

[4] J. Hoekstra, J. Ellerbroek, S. van Dam, and R. van Paassen, "Weather effects on small uas operations: Exploratory studies," *11th USA/Europe Air Traffic Management Research and Development Seminar*, 2017.

[5] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The traveling salesman problem: A computational study," *Princeton series in applied mathematics*, 2011.

[6] D. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm—a novel tool for complex optimisation problems," *Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS 2006)*, pp. 454–459, 2006.

[7] R. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, vol. 1, pp. 81–86, 2001.

[8] D. Whitley, "A genetic algorithm tutorial," *Statistics and computing*, vol. 4, no. 2, pp. 65–85, 1994.

[9] R. A. Clothier and R. A. Walker, "Risk assessment of small aircraft operations," *Proceedings of the 21st International Unmanned Air Vehicle Systems Conference*, vol. 21, pp. 21–23, 2005.

[10] A. H. Ismail, N. Hartono, S. Zeybek, and D. T. Pham, "Using the bees algorithm to solve combinatorial optimisation problems for tsplib," in *IOP conference series: materials science and engineering*, vol. 847, no. 1. IOP Publishing, 2020, p. 012027.

[11] S. Zeybek, A. H. Ismail, N. Hartono, M. Caterino, and K. Jiang, "An improved vantage point bees algorithm to solve combinatorial optimization problems from tsplib," in *Macromolecular symposia*, vol. 396, no. 1. Wiley Online Library, 2021, p. 2000299.