Adaptive Resource Allocation and Trajectory Planning Strategy for Multi-UAV Collaborative Systems

Chaitra B V, Dharamendra Chouhan, Anand R Umarji, Dilip S, Sanjay

Department of Computer Science & Engineering

University of Visvesvaraya College of Engineering (UVCE – on IIT Model) Bangalore, India

{cbelavadi21@gmail.com, dharmu2007@gmail.com, ru.anand@gmail.com, sanjaygodbole75@gmail.com }

***Abstract*— Unmanned Aerial Vehicles (UAVs), and Mobile Edge Computing (MEC), have emerged as a promising method for extending computational power to smart mobile devices in wireless communication networks that have limited battery life. UAVs that use dynamic task prioritization adjust job priorities in real-time in response to shifting mission requirements, changing environmental factors, and changing system environments. The joint optimization of the user-UAV organization, regional processing frequency, transmit authority, bandwidth allocation, and UAV trajectories maximizes the overall number of processing bits for all users while satisfying the energy consumption and speed constraints. We are obtaining the solution through genetic algorithm with pareto front optimization technique. Directly obtaining the global optimum solution is difficult as the issue is not convex. The non-convex problem is broken down into smaller issues and is then resolved by an alternating optimization approach. The approach presented in our study that outperforms fixed trajectory patterns, complete offloading, and local computing methods.**

Keywords: Unmanned Aerial Vehicles, UAV trajectories, task prioritization., resource allocation

I. Introduction

The amount of data traffic on mobile devices is increasing dramatically due to the rise in the number of wirelessly linked devices [1]. A novel computer architecture called MEC has been established to deliver low-latency and high-robust applications to end users [2]. MEC shows the benefits of reduced execution time and energy usage by harvesting the idle processing and storing ability of edge devices in a network [3], including bases stations (BSs) and access points (APs) that are wireless.

On the other hand, classic MEC systems have an inflexible and expensive deployment of MEC servers that is closely tied with ground infrastructures [4]. The use of MEC in conjunction with UAVs has emerged as a viable strategy to overcome this problem. MEC servers are mounted on UAVs, that act as floating edge computing hubs. UAV-aided MEC devices are more adaptable and simpler to set up than standard MEC systems [5]. As a result, in scenarios like disaster assistance, military training, or rescue efforts when there is little ground infrastructure, UAV-assisted MEC systems can handle the situation.
Resource allocation and task offloading algorithms have been optimized extensively in an effort to improve the efficiency of UAV-aided MEC system. Hu et al. [7], for instance, they offer both a globally and a locally optimum approach for the energy consumption reduction issue in an UAVs-aided MEC system, whereby the optimization of compute task division, communication resource allocation, and UAV deployment is combined. Similar to this, Wang et al. [8] proposed a UAV deployment method in a massive multi-UAV-aided MEC system in order to reduce the system's energy usage. The Differential evolutionary (DE) technique is used to jointly optimize the quantity and locations of UAVs. By optimizing the 3-D positions of UAVs, Sun et al. [9] developed a Sequential Polygonal Approximation (SPA) based approach to minimize the processing time of UAVs.

It is important to note that the fixed placements of the UAVs that is mentioned in [7], [8], and [9] prevent them from fully using the adaptability of UAVS. A two-stage technique for optimizing the calculation of bits in a UAV-assisted MEC wirelessly powered system is presented in [10].

In Qian et al [11] jointly optimize the user-UAV organization, UAV trajectory, and transmission power of each user equipment (UE) in order to investigate the offloading bits maximization difficulties in a UAV-based MEC system. A UAV-assisted MEC irelessly powered system is examined in [12], with the UAV flying on a predetermined path. The best user service sequences are found by using flow-shop scheduling approaches. However, in [10], [11], and [12] considered only a single UAV, which regrettably restricts the potential of UAV-assisted MEC systems.

This work investigates a multi-UAV-aided system to fully utilize numerous UAVs and their variable mobility. In this study, we jointly optimize the regional processing rate, transmit power, bits allocation, user-UAV association, and UAV trajectories in order to maximize the overall number of processing bits for all users.

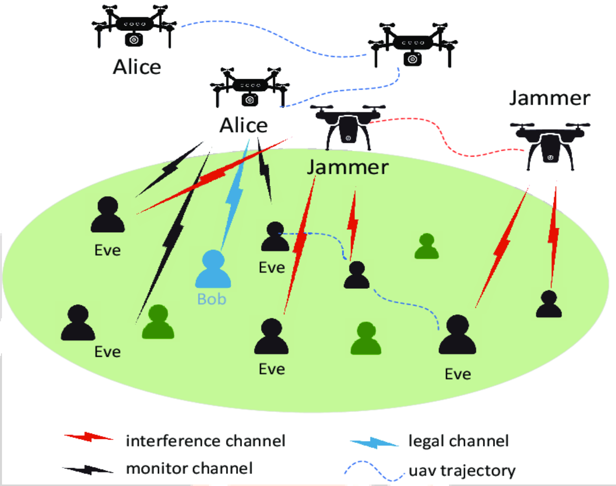

Figure 1: UAV Communication Model

The optimization is split up into smaller problems, each of which is handled using a different technique: Integer programming, Convex optimization. After that, an alternate algorithm is proposed to fix the initial issue. Based on simulations, the approach presented in our study outperforms fixed trajectory patterns, complete offloading, and local computing.

II. Literature Survey

1) Computation Offloading and Resource Allocation:

In [10], a UAV must have dependable contact with BSs throughout the slot while it is flying to a certain area for a specific mission. The objective is to minimize the UAV's completion time by 2-D trajectory optimization, by considering the BS-UAV link's connection restriction.
In [11] look at a group of UAVs cooperating, and they suggest choosing a mechanism for data distribution between UAV-to-infrastructure and UAV-to-UAV. In order to maximize the uplink data rate, allocation of resources and speed optimization are then proposed.
In [12] it is described that the secure communication is based on UAVs. One UAV is used for data transmission, and the other is used to jam the ground-based eavesdroppers in a two-UAV system. The design of UAV trajectory and user scheduling maximizes the lowest worst-case confidentiality data rate achievable for the Global unmanned system.
Regarding the 3-D orbit design, the UAV system is taken into consideration for both periodic & temporal operating modes in [13]. The goal is to reduce the amount of time that the UAV must fly or take to complete the mission in each scenario.
To achieve equitable performance, in [14] it is presented that maximizing the lowest possible throughput of each ground user (GU). The techniques for user scheduling, power allocation, and route planning are given.
In order to maximize the group's minimal throughput within a certain time frame, in [15] it is presented that a transmit allocation of power and 3-D trajectory design optimization technique. To maximize the total system usefulness, a drone-based tiny cell placement challenge is investigated in [16].
In [17] and [18], transmit power distribution and user association techniques are described to increase uplink dependability by considering the combined optimization of the UAVs' position and movement.

The work proposed in [19] is to maximize the processing speed of a solar-powered UAV system within a certain time frame by addressed the challenges of resource allocation and trajectory planning. Increasing quality of service, or QoS, is closely related to task scheduling and load balancing, and compute offloading can assist by shifting workloads to nearby MEC servers. A two-layer optimization strategy for concurrently optimizing task planning and UAV deployment is given in [21]. The highest-level optimized UAV deployment, while the bottom layer accomplished job scheduling based on the stated UAV deployment. Yang et al. [22] achieved multi-UAV load balancing while adhering to IoT node QoS requirements and coverage restrictions. Furthermore, in order to enhance the effectiveness of every UAV's task delivery, an advanced deep learning with retraining (DRL) scheduling technique was created. Researchers often see energy consumption as the optimization goal for computing and communication. Bit distribution, duration scheduling, power allocation, and UAV trajectory design were all optimized in Zhang et al. [23] by minimizing the total energy consumption (which includes interaction, computing, and UAV flight).

Furthermore, resource allocation can reduce resource waste by distributing resources to low-power ground devices fairly. In MEC systems, resource allocation and computation offloading are often combined. Seid et al. [24] proposed a model-free directional light-based cooperative allocate resources & computation offloading technique in an Ariel to Ground (A2G) network Each UAV head in the cluster acted as the agent, distributing resources among Border Network of Things sensors independently and autonomously.

Yu et al. [25] introduced an innovative UAV-enabled MEC system that combined UAVs and Edge Clouds (ECs) to provide MEC services for IoT devices. The authors' proposed system optimized UAV status, communication, task-splitting, & computer resource allocation to minimize the weighted total of service latency and UAV energy consumption for all IoT devices. Using a multiagent width dynamic guideline gradient (MADDPG) approach, Peng and Shen [8] were able to quickly determine vehicle connectivity and resource allocation during the online executing stage while managing heterogeneous QoS restrictions.

Nie et al. [9] concurrently optimized resource allocation, customers association in and power regulation in a MEC system with many UAVs and suggested a multiagent federated reinforcement learning (MFRL), technique to protect the privacy of the low-power ground devices.

Joint trajectory planning for unmanned aerial vehicles has been the subject of extensive research. The study of trajectory-planning issues for solitary vehicles is currently at a higher level of development. However, the complicated nature of combat operations frequently results in a variety of collaboration and performance limits when many UCAVs are used in cooperative missions [29]. Cheng et al. [30] defined a decentralized multi-UAV path-planning technique especially made for areas with plenty of obstacles to solve these problems. The goal of this method is to overcome the computational and scalability restrictions of conventional multi-UAV path-planning techniques. Similar to this, Liu et al. [31] examined the cooperation and flying limitations of UAVs and created a 3-D environmental model that included geographic data. The study conducted by Chen et al. [32], on the other hand, examined the dynamic and partly visible character of the surrounding conditions and accomplished k-diff multi-UAV collaborative autonomous routing in unknown situations.

Furthermore, Li et al. [33] presented a multi-drone path-planning method to overcome issues with low efficiency in proactively avoiding obstacles in a 3-D hilly terrain, lengthy planned pathways, and low stability. Additionally, a multi-UAV joint path-planning technique based on attention training was presented by Wang et al. [34]. In order to maximize multimachine collaboration, this strategy considers several aspects such as path length, load balance, endurance limits, and survival probability.

The intricacy of combat duties sometimes gives rise to various forms of cooperation and performance restrictions during execution of joint missions involving numerous UCAVs [29]. Current approaches often fall short of fully addressing these limitations, leading to trajectory failures that are not suitable for multi-UCAV collaborative warfare [35].

2) Trajectory Design:

An extensive work has been done on UAV trajectory optimisation. To lowering the latency and saving energy, it may boost communication throughput and enhance low-power ground devices for quality of service. Ji et al. [18] used a combined UAV path and resource allocation strategy to minimize the calculated energy consumption of UAVs and GDs in the 2-D plane sole UAV scenario. The authors alternatively optimized the trajectory and allocation of resources in each iteration because of the nonconvexity.

In Qin et al. [19] optimized each UAV's trajectory to minimize task completion time in the 2-D surface multi-UAV scenario while guaranteeing that all sensor's data was collected. The scientists presented a method for hover point selection, wherein UAVs systematically gathered data from several sensors.

DRL might offer a useful way to address the trajectory of the UAV. Owing to the intricate actions and vast state dimensions present in UAV communication situations, the agents in the RL framework interact with the environment to learn and "trial and error" way to the best course of action. Simultaneously, deep learning is presented as a reasonable solution to the huge data dimension problem. Yin and Yu [20] presented a novel networked multiagent RL system for overall throughput optimization and modelled the distribution of resources and trajectory planning as a decentralised partially transparent Markov decision process. By separately regulating each UAV trajectory, Wang et al. [21] simultaneously optimized the geographic unfairness of all a ground devices (GDs), the GDs-load unfairness of each UAV, including the total energy consumption of GDs. By creating a UAV trajectory, Qin et al. [22] defined a weighted throughput maximization issue and explained user-level justice based on proportional equitable scheduling.

The situation of 3-D planar multi-UAV-assisted MEC has not been extensively studied. The complexity of 3-D planar UAV actions makes it challenging to get the best answer with conventional algorithms. Few researchers are currently employing DRL to address the multi-UAV trajectory problem in three dimensions. Through trajectory design & frequency band allocation, Ding et al. [23] accomplished energy-efficient equitable communication and overall throughput maximization for a quad-rotor single UAV. The energy consumption models were developed as a function of the single UAV's 3-D motion. Effective 3-D path design for many UAVs was examined in [24]. The multi-UAV 3-D fluid motion problem was defined to be solved using a constrained deep Q-network (cDQN) approach.

In contrast, there are several papers working on 2-D trajectory design (e.g. carried out the horizontal positions) of the UAV by fixing its altitude. To address the problem of control over a group of UAVs in a long term, in [8] it utilizes the deep reinforcement learning (DRL) to minimize the energy consumption of the overall network while maintaining the reliable connectivity.

II. System Model

Figure 2 depicts the multi-UAV assisted system, in which every UAV interface with a potent MEC server. Each of the system's UAVs & M users has a strong on-board computer processor that supports computing for users while the UAV is in flight.

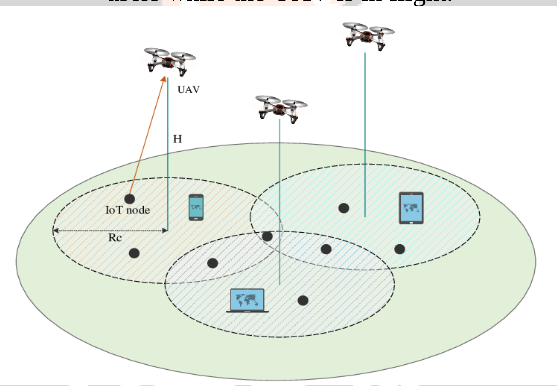

Figure 2: The model of multi-UAV-aided system

First, we design a model of the 3-D dynamic multi-UAV assisted system. Next, the system's processing and communication models as well as the UAV's flying model are executed. Finally, we frame the problem as the system's overall energy consumption including interaction, computing, and UAV flying, based on the fairness assumption for each UAV's load.

A. Task Prioritization:

Managing task arrival and prioritization is essential for maintaining the overall effectiveness and responsiveness of the system. Numerous sources, including sensors, external systems, and ground-based stations, can produce tasks.

Distinct tasks may have distinct characteristics, such as priority levels, deadlines, and computational needs. establishing a task arrival frequency depending on the tasks' type and the environment's features. Depending on the operating environment of the system, arrival rates will change.

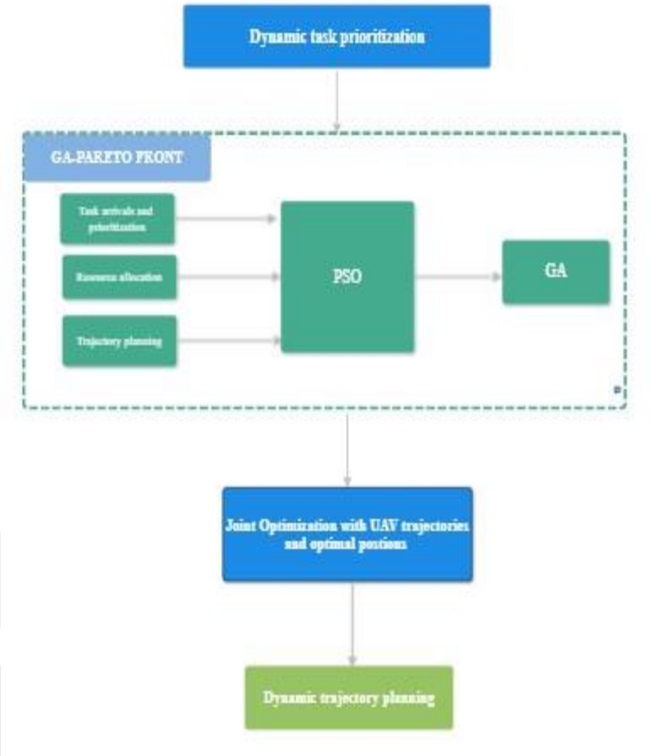

Figure 3: Dynamic Task Prioritization

Computational tasks are sorted according to their urgency, computational, complexity and criticality. Give each task a set of characteristics that will assist decide its priority. Create routes of communication so that UAVs, stations on the ground, and other relevant parties may share task information.
In multi-UAV aided systems, a dynamic task prioritization strategy should be adaptable, considering real-time task arrivals, dynamic priority modifications, and effective resource allocation. Overall in this method the maximum task performance is observed.

B. Genetic Algorithm Pareto Front Optimization

One method for resolving multi-objective optimization problem is to employ a Genetic Algorithm Pareto Front optimization function. A collection of solutions known as the Pareto Front is one in which no solution may be enhanced for one goal without impairing performance for another. The Pareto Front optimization in a Genetic Algorithm involves developing a population of potential solutions in order to identify a collection of non-dominated and varied solutions.

Since each solution in a multi objective optimization issue may maximize one objective function while sacrificing any of the others, there may be more than one solution that is deemed optimum.

A collection of solutions known as the Pareto frontier shows which trade-offs among all the goal functions are most advantageous. The Pareto frontier is a solution that is not dominated by any of the other possibilities in the possible solution space. In our work latency and throughput are normalized using this approach, so that both the parameters performance is not compromised and effective solution is achieved.

C. Working Procedure of Multi Objective Algorithms

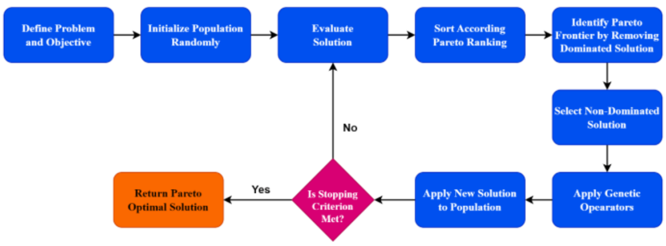

Figure 4: Pareto Front Optimization

Defining the issue and goals is the first stage. In this stage, the problem is defined as a collection of constraints and decision variables. Furthermore, we specify the goals. In addition, the goals have to be measurable in order that we can assess how well the solutions work.

Random population initialization is the second phase. We create a random population in this stage. Furthermore, the complexity of the problem affects the number of the population. The produced solutions must next be assessed. We assess the answers or view the goals. This is a crucial stage because it enables the algorithm to order the solutions according to how well they perform. The assessment procedure may incur significant computing costs. Therefore, selecting effective assessment techniques is essential.

We rate the solutions based on the Pareto ranking in order to determine which is the best option. Based on how well each solution performs for each goal, a Pareto rating is used to compare them. If no other answer in the viable solution space dominates a given solution, we refer to it as Pareto-optimal.

We determine the Pareto frontier remedies based in the Pareto ranking. Furthermore, by eliminating dominated solutions, we get the Pareto frontier solutions. Moreover, the residual solutions constitute the Pareto frontier, signifying the collection of non-dominated solutions deemed to be optimal for trade-offs. The non-dominated solutions must be chosen for replication in the following stage. We use genetic operators like crossover and mutation to execute the reproduction stage and produce new solutions.

Later, we increase the population of solutions. We now verify that the resulting solution satisfies the halting requirement. It returns the Pareto optimum solution if it meets the halting condition. If not, we begin with a new set of solutions and return to the development of the at random solution step. Further, it keeps repeating till the halting requirement is satisfied.

Simple is the Pareto dominance principle. There are two answers provided, s and s'. If at least a single of the goals is met and the remaining ones are not worse, then solution s dominates solution s'. This obviously creates a partial order, as it's possible that none of the two solutions [Gendreau, M., and Potvin, J.Y. (2010)] dominate the other. It is important that the solutions do not exhibit dominance in comparison to the solutions examined during the metaheuristic run. As a result, our set of solutions approximates the ideal Pareto front. This estimate is referred to as the estimate Pareto front in our paper. A function with an objective value can be used to determine the quality of a good solution in single-objective situations. The issue becomes hazy when we compare two approaches that are represented with estimate Pareto fronts.

An estimate of the UAVs' energy use per task input bit. (applies the PSO optimisation method)

Bit-by-bit energy consumption (E_per_bit) =

$$\frac{Total\ energy\ consumption}{Total\ Task\ input\ bits}$$

This may be expressed mathematically as E_per_bit =

$$\frac{\sum_{i=1}^{T} Ei}{\sum_{i=1}^{T} Iq1i}$$

As the mission duration and task input bit grow, the overall energy usage falls. Energy consumption for offloading (Eoff) calculates the offloading factor for user u, sum of probability of users and UAVs, data rate of user, offloading time of user, effective offloading rate of user, queuing and processing time of user u, and calculates the local time, offloading time and execution time of user u a at edge servers.

As activity input bits & operation duration rise, energy usage falls. At the maximum task intake bits and mission time values displayed in results, the eco-friendliest scenario takes place. There seems to be a non-linear relationship between mission time, activity input bits, and energy usage. A line joining data points on the results suggests a continuous link between the variables. The results do not provide specific data values, but the pattern is evident. The line slopes downhill from towards the left, illustrating a negative connection between energy usage and the sum of the variable of activity input bits & mission length. This indicates that energy usage reduces as task inputs bits and mission duration grow. Since the connection is non-linear, it is possible that there are thresholds or ideal values for the mission time and task input bits in order to maximize energy efficiency.

Subsequent investigations may explore into the fundamental processes propelling this correlation and pinpoint plausible approaches for enhancement.
UAV trajectory design & bandwidth allocation comprises planning the drones' routes and distributing communication resources to achieve predetermined goals. Often, the optimization challenge involves minimizing interference, maximizing coverage, minimizing energy use, or striking a compromise between these competing objectives.
Bandwidth allocation: Incorporates distance relationship with data rate, QoS, inference, and communication rate.

ALGORITHM: Dynamic Task prioritization

Initialize latency, throughput and energy consumption as [1,1,1]
if strcmp (algorithm, 'GA')
options = optimoptions (gamultiobj, 'PlotFcn', gaplotpareto, 'Display', 'iter');
optimizer = gamultiobj;
else if strcmp (algorithm, 'PSO')
options = optimoptions (particleswarm, 'PlotFcn', pswplotpareto, 'Display', 'iter');
optimizer = particleswarm;
else
error ('Invalid optimization algorithm. Choose "GA" or "PSO".');
end if

We have considered an objective function that calculates the objective values for the given set of parameters. Declare the objectives and set options for optimization algorithm. Taken an array containing optimized parameter values, ignored values (function handles excluded) and information about optimization process.
The working involves defining optimization problem which minimizes multi-objective function. Run genetic algorithm (GA) using optimizer function gamultiobj, pass the objective function and other parameters as input. Store the optimized parameter values in a variable and ignore the additional values if they are not used in this context. If necessary, you can examine the details about the optimization process found in the output structure.

V. EXPERIMENT RESULTS

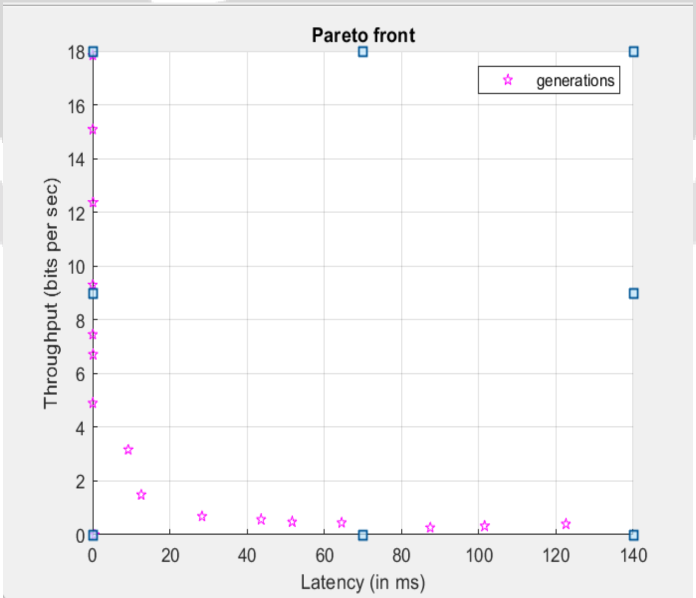

Figure 5: Dynamic task prioritization

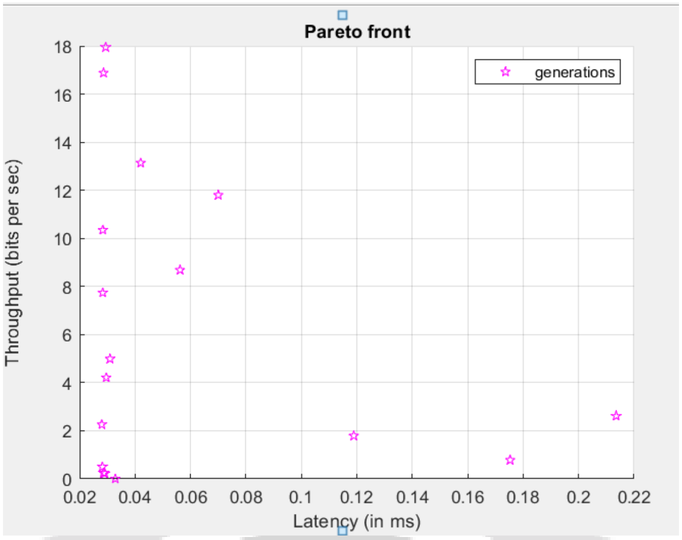

Figure 6: Dynamic task prioritization using Pareto front.

Figure 5 and 6 represents the outcomes of a multi-objective optimization problem using a Genetic Algorithm (GA). Here, latency i.e., the amount of time it takes to do a task and throughput i.e., the speed at which tasks are finished are the two factors that the GA aims to minimize.

The Pareto front, or collection of optimal solutions that are not dominated by any other solution, is represented by the stars in the graph. In other words, there isn't another solution that possesses both reduced latency and increased throughput for any given Pareto front solution.

Pareto front optimization function using evolutionary algorithm to normalize latency, throughput, and energy consumption often, there isn't a single "best" solution in multi-objective optimization scenarios including genetic algorithms.
Rather, we possess a collection of solutions that embody the optimal compromises among several competing goals. The Pareto front is the name of this group of solutions.

This set of answers is represented graphically in a space with several dimensions by a Pareto front graph. Every dimension denotes an objective function that is undergoing optimization. Pareto-frontal solutions are regarded as non-dominated. One goal function is represented by each axis. There are two axes in a 2-dimensional graph, which are often labelled with the initials of the objectives i.e., latency and throughput. More values on an axis typically indicate better achievement of that goal. Every point on the graph is a potential solution that the genetic algorithm was able to identify. The values of the point for all objective functions define its location in space.

The shape or surface created by solutions that don't dominate is known as the Pareto front. It shows the range of potential compromises between the goals without compromising on effectiveness.

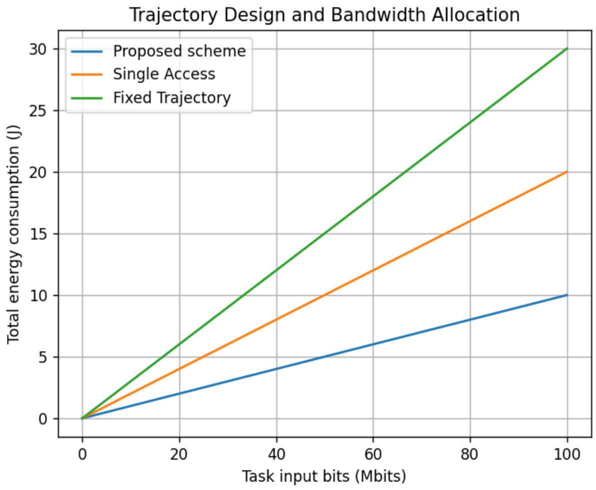

Figure 7: Trajectory design and Bandwidth allocation

Figure 7: Shows the energy consumption throughout the course of the mission across the task input bits (total energy consumption decreases during the mission length). Also, the graph represents total energy consumption decreases during the mission length.

X-axis- Task input bits (Mbits) this likely represents the amount of data or information each UAV needs to handle on its assigned task. Y-axis - total energy consumption (J) this represents the total energy each UAV consumes during its assigned task. The proposed scheme consistently has the lowest energy consumption across all task input bit values. This shows that the proposed scheme is more efficient in terms of energy usage compared to single access and fixed trajectory approaches. By employing collaboration and dynamically adjusting trajectories and bandwidth allocation based on task requirements, the proposed scheme can significantly reduce energy consumption compared to traditional single access and fixed trajectory approaches.

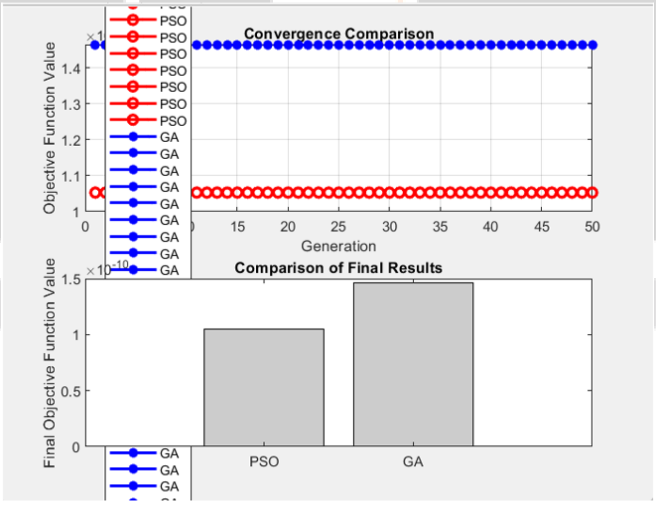

Figure 8: PSO and GA comparison results

In the cases when GA outperforms PSO, GA also exhibits higher convergence towards the ideal solution than PSO. Fig 8 showing the convergence rate of PSO and GA algorithms for an optimization problem. X-axis represents the generation or iteration number, Y-axis represents the objective function value, with lower values indicating better solutions. In GA convergence, GA line appears to have a steeper initial descent, presents it finds better solutions faster in the early stages. The PSO line has a slower initial descent but seems to eventually reach a lower objective function value than GA.

A comparison between GA and PSO across all displayed data points, GA looks to be more environmentally friendly than PSO. It appears that the two algorithms' differences in energy usage are greater for lower beliefs of task intake bits and mission duration. In addition, it appears that GA converges faster to the best solution (lowest energy usage) than PSO did.

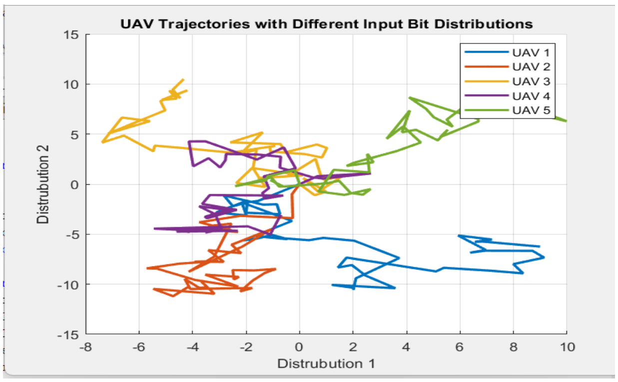

Figure 9: UAV Trajectories with different input distributions

X-axis represents the number of task input bits, ranging from 0 to 10. Y-axis represents the probability density, which indicates the relative frequency of a certain number of input bits occurring. Blue line represents the uniform distribution. In this case, all values between 0 and 10 have an equal probability of occurring. This means the line is flat across the X-axis. Green line represents the normal distribution. The peak of the curve is at the mean value of the distribution, and the probability of input bit values decreases as you move away from the mean in either direction. Orange line represents the Poisson distribution. The peak of the curve is at 0 input bits, and the probability of input bit values decreases as the number of bits increases. Overall, this provides a helpful visualization of the different probability distributions that can be used to model UAV task input bits.

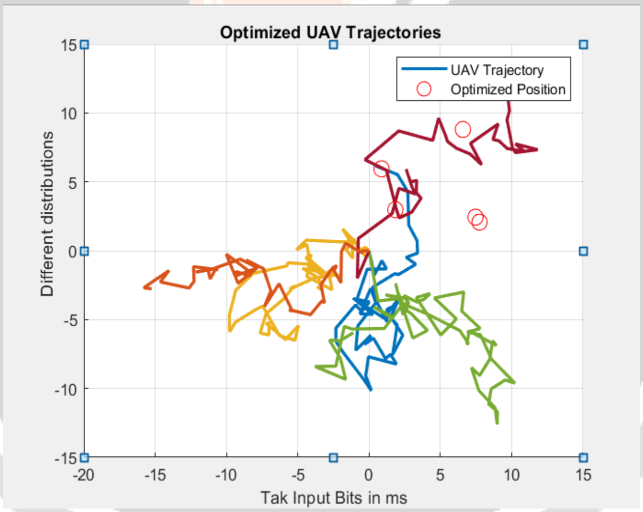

Figure 10: Optimized UAV trajectories

Fig 10 represents the UAV trajectories with optimal positions where X axis represents the task input bits and Y axis represents different distrubutions with respect to UAV trajectories. The optimal positions shows that they have achieved the normalized latency and throughput with the proposed multi-objective algorithm the red lines, with optimized positions indicating an optimized UAV trajectory. The lines indicate distinct beginning and finishing sites for each trajectory, with their starting points on the left & endings on the right. The lines are curved and smooth, showing effective routes that get around obstructions and other limitations.

In figure 11 messages showing whether the IoT or UAV device falls within the communication spectrum and the data each device has received will be included in the simulation output. If the devices are not within communication range, this will be displayed in output. This offers a fundamental modelling framework for comprehending situations of data transfer in a basic 3-D environment between a UAV as an IoT device.
The negative value in the graph shows that the data transfer is between UAV and IoT from the Z-axis in a 3D representation. This is shown in Fig 13. Also, negative value in data transfer indicates which means that the UAVs positioned in Z axis are communicating with IoT device. If UAVs are not under communication range, then a message is displayed on the output screen.

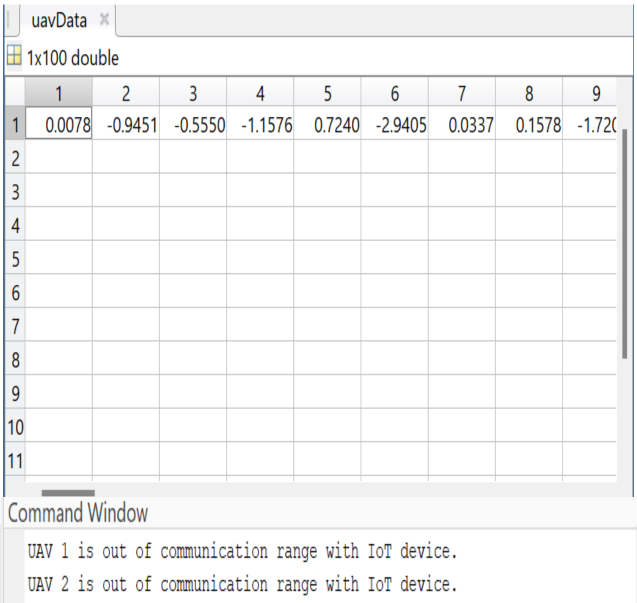

Figure 11: Data transfer between UAV and IoT

In figure 12 messages showing whether the IoT or UAV device falls within the communication spectrum and the data each device has received will be included in the simulation's output. If the devices are not within communication range, this will also be shown by the output. This offers a fundamental modelling framework for comprehending situations of data transfer in a basic 3-D environment between a UAV as an IoT device.

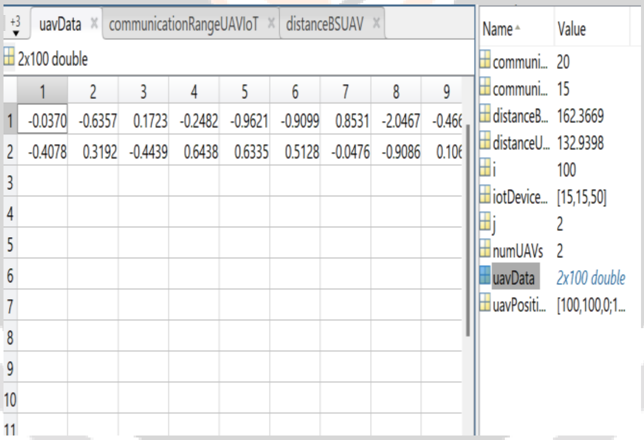

Figure 12: Figure 12: Communication between base station, UAV and IoT

Displayed messages indicating whether every UAV falls within communication range of the base station and the IoT device will be included in the output. Additionally, it displays the data that each UAV receives from the central station as well as the data that each IoT device receives from each UAV. In order to keep things simple, this simulation uses random data. The IoT device and UAVs are assumed to travel in straight lines across time increments. The ability of devices to communicate with one another is determined by the communication ranges. In order to comprehend communication situations including a centre station, UAVs, & an IoT item in a 3D space, this offers a fundamental simulation framework.

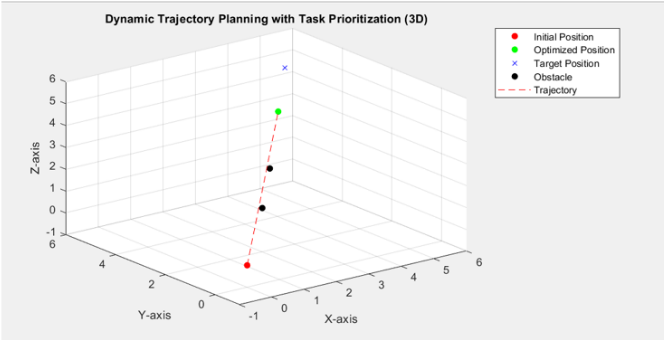

Figure 13: Dynamic 3D planning

Figure 13 shows dynamic 3-D planning in 3-D space. The line passes the obstacles and reaching the optimized position so that it can reach the target position without any restrictions, this ensures goal optimization. The dynamic planning including task prioritization for obstacle avoidance within 3-D space is shown in this graph. The system under consideration has barriers, a goal position, and a beginning position. Finding the optimal location that satisfies restrictions and minimizes a cost function is the goal. The result consists of a 3-D graphic of the situation and the optimized position. This illustrates a basic dynamic planning for trajectory scenario in which the system seeks to avoid obstacles and achieve an objective. By giving various tasks varying weights, tasks may be prioritized. Finding the ideal system position that strikes a balance between achieving the goal and minimizing impediments is the main goal of optimization.

IV. Conclusions

In this research, we maximize the overall number of processing bits per user for multi-UAV assisted systems. The non-convex computing bits maximization problem is approached by splitting it into smaller problems, and using our suggested altering optimization technique, a locally optimum solution is found. Thereby achieving less energy consumption in the system. By employing dynamic task prioritization with GA and PSO simulations showed that, within the energy constraint, the approach presented in our study outperforms the fixed trajectory, localized computing, and complete offloading patterns. Also, this shows the obstacle avoidance in 3-D space with pareto optimization technique which minimizes the cost function. Also, by considering the use case scenario the work shows the communication between base station, UAV and IoT device and data transfer between UAV and IoT device.

References

[1] Yastrebova A, Kirichek R, Koucheryavy Y, Borodin A and Koucheryavy A “Future networks 2030: Architecture & requirements Proc. 10th Int. Congr. Ultra-Mod. Telecommand. Control Syst”. Workshops (ICUMT) Nov 2018 pp. 1–8.
[2] Chiang M and Zhang T Fog and IoT: An overview of research opportunities Dec. IEEE Internet Things J. 2016 3 854–64
[3] Mao Y, You C, Zhang J, Huang K and Letaief K B A survey on mobile edge computing: The communication perspective IEEE Commun. Surveys Tuts. 19 2322–58
[4] Zhang W, Li L, Zhang N, Han T and Wang S 2020 Air-Ground Integrated Mobile Edge Networks: A Survey IEEE Access 8 125998–6018
[5] Choi D H, Kim S H and Sung D K, Jul. 2014 Energy-efficient maneuvering and communication of a single UAV-based relay IEEE Trans. Aerosp. Electron. Syst. 50 2320–27
[6] Nguyen V, Khanh T T, Van Nam P, Thu N T, Seon Hong C and Huh E 2020 Towards Flying Mobile Edge Computing 2020 Int. Conf. on Information Networking pp. 723–5
[7] Hu J, Jiang M, Zhang Q, Li Q and Qin J July 2019 Joint Optimization of UAV Position, Time Slot Allocation, and Computation Task Partition in Multiuser Aerial Mobile-Edge Computing Systems IEEE Trans. Veh. Technol. 68 7231–35
[8] Wang Y, Ru Z Y, Wang K and Huang P Q Sept. 2020 Joint Deployment and Task Scheduling Optimization for Large-Scale Mobile Users in Multi-UAV-Enabled Mobile Edge Computing IEEE T. Cybern. 50 3984–97
[9] Sun S, Zhang G, Mei H, Wang K and Yang K Feb. 2021 Optimizing Multi-UAV Deployment in 3-D Space to Minimize Task Completion Time in UAV-Enabled Mobile Edge Computing Systems IEEE Commun. Lett. 25 579–83

[10] S. Zhang, Y. Zeng, and R. Zhang, "Cellular-enabled UAV communication: A connectivity-constrained trajectory optimization perspective", IEEE Transactions on Communications, vol. 67, no. 3, pp. 2580-2604, Mar. 2019.
[11] S. Zhang, H. Zhang, B. Di, and L. Song, "Cellular UAV-to-X communications: Design and optimization for multi-UAV networks", IEEE Transactions on Wireless Communications, vol. 18, no. 2, pp. 1346-1359, 2019.
[12] Y. Cai, F. Cui, Q. Shi, M. Zhao, and G. Y. Li, "Dual-UAV-enabled secure communications: Joint trajectory design and user scheduling," IEEE Journal on Selected Areas in Communications, vol. 36, no. 9, pp. 1972-1985, Sept. 2018. [13] J. Zhang, Y. Zeng and R. Zhang, "UAV-Enabled radio access network: multi-mode communication and trajectory design," IEEE Transactions on Signal Processing, vol. 66, no. 20, pp. 5269-5284, 15 Fig. 9. The impact of the number of UAVs on system throughput Oct.15, 2018.
[14] Q. Wu, Y. Zeng and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," IEEE Transactions on Wireless Communications, vol. 17, no. 3, pp. 2109-2121, March 2018.
[15] H. Wang, G. Ren, J. Chen, G. Ding, and Y. Yang, "Unmanned aerial vehicle-aided communications: Joint transmit power and trajectory optimization," IEEE Wireless Communications Letters, vol. 7, no. 4, pp. 522-525, Aug. 2018.
[16] R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, "Efficient 3- D placement of an aerial base station in next generation cellular networks," Proc. IEEE Int. Conf. Commun. (ICC), May 2016.
[17] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient internet of things communications," IEEE Transactions on Wireless Communications, vol. 16, no. 11, pp. 7574-7589, Nov. 2017.
[18] Z. Yang et al., "Joint altitude, beamwidth, location, and bandwidth optimization for UAV-enabled communications," IEEE Communications Letters, vol. 22, no. 8, pp. 1716-1719, Aug. 2018.
[19] X. Yuan, Y. Hu, D. Li and A. Schmeink, "Novel optimal trajectory design in UAV-assisted networks: A mechanical equivalence-based strategy," IEEE Journal on Selected Areas in Communications, vol. 39, no. 11, pp. 3524-3541, Nov. 2021
[20] Zhou F, Wu Y, Hu R Q and Qian Y Sept. 2018 Computation Rate Maximization in UAVEnabled Wireless-Powered Mobile-Edge Computing Systems IEEE J. Sel. Areas Commun. 36 1927–41
[21] Qian Y, Wang F, Li J, Shi L, Cai K and Shu F Oct. 2019 User-UAV association and Path Planning Lett. 8 1312–15
[22] Du Y, Yang K, Wang K, Zhang G, Zhao Y and Chen D Oct. 2019 Joint Resources and Workflow Scheduling in UAV-Enabled Wirelessly-Powered MEC for IoT Systems IEEE Trans. Veh. Technol. 68 10187–200
[23] Zeng Y, Zhang R and Lim T J May. 2016 Wireless communications with unmanned aerial vehicles: Opportunities and challenges IEEE Commun. Mag. 54 36–42
[24] Cui M, Zhang G, Wu Q and Ng D W K Sep. 2018 Robust trajectory and transmit power design for secure UAV communications IEEE Trans. Veh. Technol. 67 9042–46
[25] Zhang X, Zhong Y, Liu P, Zhou F and Wang Y 2019 Resource Allocation for a UAV-Enabled Mobile-Edge Computing System: Computation Efficiency Maximization IEEE Access 7 113345–54
[26] Wang Y et al. Oct. 2016 Mobile-edge computing: Partial computation offloading using dynamic voltage scaling IEEE Trans. Commun. 64 4268–82
[27] Lodi A 2010 Mixed integer programming computation Years Integer Program. 619–45
[28] Mehrotra S 1992 On the implementation of a primal-dual interior point method Siam J. Optim. 2 575–601
[29] Smith, J.; Johnson, A.; Brown, R. Multi-UAV cooperative mission planning considering collaborative and performance constraints. J. Intell. Syst. 2020, 45, 567–582. [Google Scholar]
[30] Cheng, Z.; Zhao, L.; Shi, Z. Decentralized Multi-UAV Path Planning Based on Two-Layer Coordinative Framework for Formation Rendezvous. IEEE Access 2022, 10, 45695–45708. [Google Scholar] [CrossRef]
[31] Liu, X.; Su, Y.; Wu, Y.; Guo, Y. Multi-Conflict-Based Optimal Algorithm for Multi-UAV Cooperative Path Planning. Drones 2023, 7, 217. [Google Scholar] [CrossRef]
[32] Chen, Y.; Dong, Q.; Shang, X.; Wu, Z.; Wang, J. Multi-UAV Autonomous Path Planning in Reconnaissance Missions Considering Incomplete Information: A Reinforcement Learning Method. Drones 2023, 7, 10. [Google Scholar] [CrossRef]
[33] Li, S.; Zhang, R.; Ding, Y.; Qin, X.; Han, Y.; Zhang, H. Multi-UAV Path Planning Algorithm Based on BINN-HHO. Sensors 2022, 22, 9786. [Google Scholar] [CrossRef]
[34] Wang, T.; Zhang, B.; Zhang, M.; Zhang, S. Multi-UAV Collaborative Path Planning Method Based on Attention Mechanism. Math. Probl. Eng. 2021, 2021, 6964875. [Google Scholar]
[35] Chen, Q.; Zhang, G.; Liang, B.; Wang, Y. A hybrid heuristic algorithm for multi-UAV cooperative trajectory planning problem. IEEE Trans. Syst. Man Cybern. Syst. 2019, 49, 1479–1492. [Google Scholar]