Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Autonomous UAV Path Planning using Modified PSO for UAV-Assisted Wireless Networks

AMALA SONNY, SREENIVASA REDDY YEDURI, (Member, IEEE), and LINGA REDDY CENKERAMADDI, (Senior Member, IEEE)

ACPS Research Group, the Department of Information and Communication Technology, University of Agder, Norway (e-mail: amalasonny@gmail.com, sreenivasa.r.yeduri@uia.no, linga.cenkeramaddi@uia.no).

Corresponding author: Linga Reddy Cenkeramaddi (e-mail: linga.cenkeramaddi@uia.no).

This work was supported by the Indo-Norwegian Collaboration in Autonomous Cyber-Physical Systems (INCAPS) project: 287918 of the International Partnerships for Excellent Education, Research and Innovation (INTPART) program and the Low-Altitude UAV Communication and Tracking (LUCAT) project: 280835 of the IKTPLUSS program from the Research Council of Norway.

ABSTRACT Recently, unmanned aerial vehicles (UAVs) have attained considerable attention for providing reliable and cost-effective communication due to the flexibility of deployment and line of sight (LoS) propagation. Efficient UAV path planning is one of the key aspects that need to be addressed to minimize energy consumption and satisfy the rate requirements of the user. Thus, in this work, we propose a novel framework that utilizes the modified Particle Swarm Optimization (PSO) algorithm for UAV path planning to support the rate requirements of the user. In the proposed framework, the problem of joint path planning and energy consumption is formulated to improve the instantaneous sum rate of the user. In order to solve the formulation, the proposed framework involves two steps. Initially, the line of sight probability is used to obtain an optimal destination location at which the UAV is in LoS with the user and offers the required downlink rate. Following that, the modified PSO is used to find the most energy-efficient path from the source to the destination. Through experiments, we show that the proposed framework provides a three-dimensional (3D) path in a complex environment, and has the ability to avoid obstacles in the path. In addition, it minimizes energy consumption and travel time and improves the user rate as compared to the state-of-the-art methods. Finally, the performance of the framework is tested in three different scenarios.

INDEX TERMS Energy consumption, Line of Sight (LoS) propagation, LoS probability, Obstacle avoidance, Path planning, Particle Swarm Optimization (PSO), Sum Rate, and Unmanned aerial vehicles (UAVs).

I. INTRODUCTION

UNMANNED aerial vehicles (UAVs) have been proposed for a wide range of applications that include surveillance, mapping, tracking, and search operations [1], [2]. Due to the flexibility of deployment, UAVs are used to collect data from remote locations. The recent availability of low-cost UAVs drives the usage of multiple UAVs for the same. Thus, efficient methods of decentralized sensing and cooperative path planning are necessary to leverage the capabilities of the group of UAVs [3]. Hence, reliable and reasonable path planning can ensure the safety of the UAV and also the success of the mission in a complex environment. However, there are many constraints for UAV path planning that include limited resources and restrictions on flight altitude. In low altitudes, there are various obstacles in the environment such as buildings, trees, and hills. Due to the large state space, path planning for UAVs is more complicated than the path planning of robots [4]. Moreover, the huge number of action states makes the process of path planning complex and slowly converging. Due to the limited sensor capacity, the UAV can only attain very limited information on the local environment. This makes it difficult for the UAV to select the optimum path. In addition to that, in large environments, the path planning algorithm might have to estimate a huge number of steps which might lead to significant delays. These are some major challenges of path planning for UAVs in a complex and dynamic environment.

To tackle these challenges, nature-inspired approaches are being adopted nowadays due to their capability to explore the global optimum path in complex environments. Algorithms such as cuckoo search [5], genetic algorithm [6], particle swarm optimization (PSO) [6], [7], artificial bee colony [8], and ant colony optimization [9] are highly effective in dealing with UAV dynamic constraints and limitations. [10] explores the construction of an integration and genetic algorithm (GA)-based path planning algorithm for an autonomous UAV in target coverage problems. The initial approach involves employing a classical GA. To address altitude limitations to prevent unsafe paths that could lead to terrain surface collisions, three different approaches are integrated into the initial population phase of the genetic algorithm. Among these approaches, PSO is a population-based algorithm which has two key properties of swarm intelligence such as cognitive and social coherence [11]. These properties enable each particle in the swarm to search for the solution based on its own experience and the swarm's experience. Thus, PSO is capable to reach the global solution in a short time with a stable convergence than other nature-inspired algorithms. In addition to this, PSO is less sensitive to initial conditions and more adaptable to changes in the objective functions [12]. It is also easier to adopt various environmental structures by using acceleration coefficient and weight factors [13]. In [14], another PSO-based framework named Sparrow Particle Swarm Algorithm (SPSA) has been proposed for UAV path planning. It improves path initialization, updates discoverer positions, and emphasizes the start-end line's influence to reduce the blind search. Adaptive variable speed escapes enhance the target to reach deadlocked areas, while adaptive oscillation optimization minimizes path fluctuations. Simplification and interpolation with cubic splines improve path smoothness, making it suitable for real flight trajectories. The improved SPSA demonstrates fast convergence, better search results, and avoidance of local optima.

Motivated by the above discussion, in this paper, we propose a novel framework that utilizes the modified Particle Swarm Optimization (PSO) algorithm for UAV path planning to support the rate requirements of the user. The major contributions of the proposed work are as follow:

- We propose a novel framework for UAV path planning in order to provide the user with downlink services. We formulated the problem as a joint optimization problem in terms of path planning and energy consumption, with the user's required sum rate as a constraint.
- The line of sight probability is used to find an optimal destination location where the UAV is in LoS with the user and can provide the required downlink rate.
- The modified PSO algorithm is used to find the most energy-efficient path from the source to the destination.
- The framework's performance is evaluated in three scenarios: i) Scenario I considers a 2D space where the height of the UAV is constant throughout the journey till the target point; ii) Scenario II deals with a 3D space where the UAV starts from the ground and reaches the target at a height h with obstacles with less height in the vicinity of the UAV flight path; and iii) Scenario III deals with the same 3D scenario as scenario II along with the presence of much longer obstacles in the path of the UAV. This is to evaluate the performance of the proposed algorithm in the presence of obstacles.
- Through experimental results, we show that the proposed framework provides a three-dimensional (3D) path in a complex environment, and has the ability to avoid obstacles in the path. In addition, it minimizes energy consumption, travel time and improves the user rate as compared to the state-of-the-art methods. Finally, the performance of the framework is tested in three different scenarios.

The remainder of the paper is organized as follows. The related works are discussed in Section II. The system model and problem formulation are described in Section III. The proposed particle swarm optimization method is elaborated in Section IV. The performance of the proposed method is evaluated in Section V. Finally, Section VI provides concluding remarks with possible future works.

II. RELATED WORKS

A survey on 3D path planning algorithms for UAVs has been presented in [15] that classified the path planning algorithms into five classes such as node-based algorithms, Bio-inspired algorithms, mathematical model-based algorithms, sampling-based algorithms, and multi-fusion-based algorithms [15]. A survey on the computational intelligence (CI) methods has been presented in [16] for UAV path planning. This survey presents an overview of different CI path planning methods with a focus on time domain-based methods such as offline and online for both 2D and 3D environments [16]. In [17], a survey on UAV path planning algorithms that includes Artificial Potential, RRT, D-Star, Voronoi, Dijkstra, A-Star, and Neural Network has been presented with a focus on the applications such as disaster management, communication networks, and general. It has been mentioned that among the UAV path planning algorithms 67% focused on UAV path planning for general applications, 22% focused on disaster management, and only 11% focused on communication networks. Finally, the authors have recommended a holistic IoT-powered UAV-based smart city management system, that integrates all the smart city components, for addressing earthquakes, disasters, and bush fire [17]. A novel optimization algorithm namely a multi-frequency vibrational genetic algorithm (mVGA) has been proposed in [18] for UAV path planning in two different 3D environmental models such as city and sinusoidal. The proposed mVGA algorithm emphasizes local random and global random diversity as the new mutation application strategy and diversity variety. Further, the initial population phase of the algorithm uses the Voronoi diagram and Clustering method concepts. Finally, it has been shown in [18] that mVGA algorithm results in reduced computational time when compared to three other genetic algorithms.

PSO plays a crucial role in UAV path planning due to its effectiveness in addressing optimization problems [7]. When compared with other path planning algorithms, PSO algorithm has several advantages. Firstly, it has a fast convergence speed compared to other optimization algorithms, making it suitable for real-time path-planning applications. Secondly, it is a global optimization algorithm that can search for the optimal path in a large search space, making it effective for complex path-planning problems. Thirdly, PSO algorithm is a robust algorithm that can handle noisy fitness functions and dynamic environments, making it suitable for real-world applications. Next, it has a simple structure and is easy to implement, making it accessible for researchers and practitioners with limited resources. Finally, it can be easily adapted to different types of path-planning problems by changing the fitness function or adjusting the acceleration coefficients, making it a versatile algorithm. These above-mentioned advantages make PSO algorithm a powerful and simple tool for path-planning applications, especially in robotics, unmanned aerial vehicles, and mobile robots. Inspired by these advantages, a spherical vector-based particle swarm optimization (SPSO) approach is proposed in [19]. The path length-based cost function formulated is to convert the path planning into an optimization problem which incorporates all the constraints for a feasible and safe operation of the UAV. It is developed on the relationship between UAV intrinsic motion components and the search space. An improved PSO algorithm specifically designed for real-time UAV path planning is proposed in [20]. The proposed algorithm incorporates a Chaos strategy to prevent particles from converging to local optima, while the Dijkstra algorithm is employed to enhance the path quality. A path planning approach that considers the dynamic properties of the UAV and the complexity of a real 3D environment is presented in [6]. They have used a combination of two nondeterministic algorithms such as Genetic Algorithm (GA) and PSO, to solve the complexity and longer computation time. Another improved PSO algorithm called GBPSO is proposed in [21] to enhance the performance of three-dimensional path planning for fixed-wing UAVs. The GBPSO algorithm aims to improve both the convergence speed and searchability of particles by incorporating a competition strategy into the standard PSO framework. In this approach, during the particle evolution process, the competition strategy optimizes the global best solution. This is achieved by considering two candidate global best paths. The first candidate path is obtained by selecting the optimal path found through a single waypoint selection approach, guided by a set of segment evaluation functions. The second candidate path is generated based on the particle as an integrated individual, representing an optimal trajectory from the start point to the flight target. [22] introduces a method for addressing the path planning problem for unmanned aerial vehicles (UAVs) operating in adversarial environments, which include radar-guided surface-to-air missiles (SAMs) and unknown threats. The approach takes into account the SAM lethal envelope and radar detection for SAM threats, as well as LOS calculations for unknown threats, to compute the path planning cost. To find an optimal path, an improved PSO algorithm is adopted along with preprocessing steps. The preprocessing steps involve generating a Voronoi diagram and using the Dijkstra algorithm to obtain an initial path for the multi-swarm PSO algorithm. The multi-swarm PSO algorithm employs multiple swarms with sub-swarms to balance exploration and exploitation. A three-dimensional path planning algorithm for UAV formation using an enhanced PSO approach is proposed in [23]. The objective of this work is to improve the speed and optimality of the automatic path planner. Hence, it incorporated several improvements to the PSO algorithm. Firstly, a chaos-based Logistic map is employed to enhance the initial distribution of particles, leading to better exploration of the solution space. Additionally, the commonly used constant acceleration coefficients and maximum velocity are replaced with adaptive linear-varying ones. This adaptation allows the coefficients to adjust during the optimization process, resulting in improved solution optimality.

However, the PSO algorithm does have a few limitations. The performance of PSO can be highly sensitive to the choice of its parameters, such as the acceleration coefficients and the inertia weight. Opting parameter tuning techniques, such as grid search, genetic algorithms, or other optimization methods, can be employed to find suitable parameter values for a specific problem [24]. Additionally, PSO can sometimes converge to a suboptimal solution prematurely, especially when dealing with complex and multimodal problems. This happens when particles cluster around a local optimum, preventing the exploration of other potential regions in the search space. Introducing diversity maintenance techniques such as adding random perturbations to particle positions can be a solution to this premature convergence problem [25]. When dealing with high-dimensional optimization problems or if particles in the swarm are large, PSO becomes computationally expensive. The computational cost of evaluating the fitness function for each particle will be directly proportional to the size of the search space. To mitigate this, parallelization techniques can be employed, where multiple particles can be evaluated concurrently using parallel computing resources. Additionally, various dimensionality reduction techniques can be applied to reduce the number of variables. Another drawback of PSO is it can get trapped in local optimas, especially in complex problems with multiple local optima and a single global optimum. Using population diversity measures and restart strategies can increase the chances of escaping local optima and finding better solutions [26].

Here, to establish a reliable communication link between the UAV and the user, the normal PSO algorithm set the target position at an LoS point from the user. This is not essential in most of the cases which leads to unwanted energy and time consumption by the UAV. Hence, in this work, we propose a modified PSO framework for UAV path planning in order to provide the user with a reliable downlink service even before reaching the LoS point with the user. It utilizes the LoS prob-

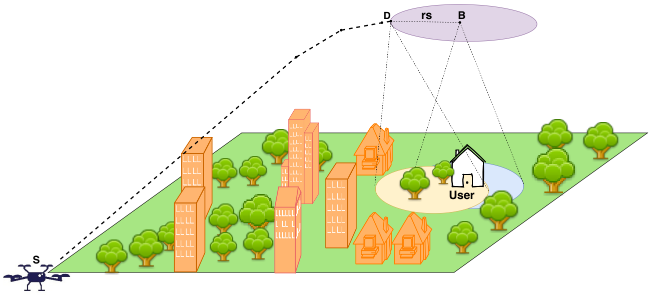


FIGURE 1: An illustration of the UAV path planning and downlink transmission.

ability to find an optimal destination location where the UAV is in LoS with the user and can provide the required downlink rate. The proposed modified PSO algorithm checks whether the required rate is achieved after each particle update as the stopping condition. In PSO, typically the target fitness value represents the distance to the target position or proximity to obstacles. By adopting this modified target fitness value, the algorithm minimizes the energy consumption, and travel time of the UAV and improves the user rate as compared to the state-of-the-art methods. A detailed description of the proposed approach is given in the following sections.

Table 1 describes the important notations used in this paper.

III. SYSTEM MODEL AND PROBLEM FORMULATION

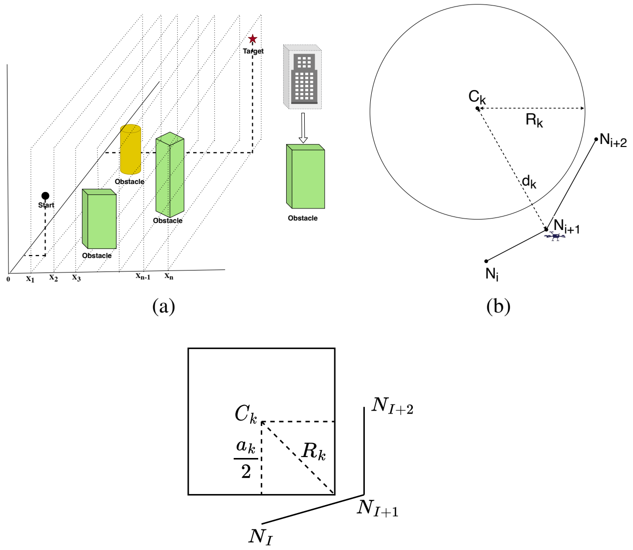
We consider a UAV-assisted wireless network scenario wherein, a UAV is deployed to offer downlink services to a user as shown in Fig. 1. Here, we define the start point as the point at which the UAV collects the data and the endpoint is the point at which there is LoS UAV-user downlink. Further, an urban scenario with multiple obstacles such as trees, buildings, and hills are considered in the UAV path. A flight path X is represented by several intermediate points through which the UAV flies. Each of these nodes corresponds to a path node in the proposed PSO search map with coordinates (x_i, y_i, z_i). In order to establish an ideal communication link between the UAV and the user, the target point of the UAV should be at the LoS point. Instead of setting the target position at a Line of Sight (LoS) point from the user, a modified point is estimated here. This modification aims to provide the user with a reliable downlink service even before reaching the LoS point. The modified approach takes into account the LoS probability to determine an optimal destination location where the UAV is within a reliable distance from the user and can deliver the required downlink rate. Here, we assume the UAV is flying with a constant speed and then the objective is a simplified line planning problem [7]. When a new position in x and y dimension is selected, the algorithm updates the z position by one unit forward. This converts the whole 3D path planning into a simple 2D one. We assume that the UAV moves with a constant speed ν. Next, we present the modeling of the environment that contains multiple obstacles.

TABLE 1: Summary of important notations used in this work.

Notation	Description
c_1,c_2	learning factors
w	Inertia weight
r_1, r_2	random numbers from [0,1]
x_{UAV}, y_{UAV}	Coordinates of UAV
x_{user}, y_{user}	Coordinates of user
N_0	Noise Power Spectral
μ_{LoS}	Attenuation factor corresponding to LoS link
μ_{NLoS}	Attenuation factor corresponding to NLoS link
$r(t)$	Achievable rate at time t
g	Channel Power gain
$h(t)$	Altitudde of UAV at time t
α	Path loss exponent
B	Bandwidth
P_{LoS}, P_{NLoS}	LoS and NLoS Probability
I	Receiving interference
τ	P_{LoS} Threshold
E_0, E_1, E_2	Energy scaling parameters
μ_0	Turning angle coefficient
$b1, b2$	Environmental impact factors
ζ	Environment factor
Γ	SINR
μ_0	Turning angle coefficient
d	distance between nodes
f_c	Carrier frequency
c	Speed of electromagnetic wave
θ_e	The elevation angle between UAV and user
V_i	Velocity of the particle
X_i	Position of the particle
Ψ_i	Best position of the i^{th} particle
Ψ_g	Best position of population
v_{max}	Maximum allowed velocity of a particle
x_{min}, x_{max}	Lower and upper boundary values of the solution space
M	Population size
x_n, y_n, z_n	Coordinates of the path node
F	Cost Function
C_k	Center of the k^{th} obstacle
R_k	Radius of the k^{th} obstacle

A. ENVIRONMENT MODEL

In UAV path planning, the main challenge is to ensure the safe operation of a UAV in a dynamic and complex environment. Major threats are caused by obstacles, such as trees, hills, and buildings, present in the terrain of UAV flight. Modelling these obstacles is a challenging task in path planning. To simplify the modeling process and computation complexity, we adopt the model proposed in [7]. In [7], each



(a) (b)

(c)

FIGURE 2: Environmental model.

obstacle k is assumed to be a cylinder or a square obstacle as shown in Fig. 2a. The cylinder projection has the center coordinate C_k and radius R_k as shown in Fig. 2b and the square projection has center coordinate C_k and a side length of a_k as shown in Fig. 2c. When a new position in x and y dimension is selected, the proposed path planning framework replans the new position in case the selected position appears to be the obstacle position. The algorithm takes into account the presence of square obstacles and dynamically checks their boundaries to ensure safe and efficient path planning. If the planned path intersects with the boundaries of the obstacles, the algorithm triggers a replanning process to find an alternative path that avoids the obstacles. This feature enhances the algorithm's ability to navigate through environments with square obstacles effectively. Nevertheless, this algorithm was specifically developed for urban areas with a higher concentration of buildings. Its effectiveness and efficiency in rural areas remain unexplored and require further investigation.

B. TRANSMISSION MODEL

In this model, the downlink between the UAV and the user can be regarded as air-to-ground communication. The LoS and Non-Line-of-Sight (NLoS) conditions are assumed to be encountered randomly. For the UAV-user link, the LoS probability is given by [27]

$$P_{LoS}(\theta_t) = b_1 \left(\frac{180}{\pi} \theta_t - \zeta \right)^{b_2} \qquad (1)$$

where, $\theta_t = \sin^{-1}(h(t)/d(t))$ is the elevation angle between the UAV and the user at time instant t. $h(t)$ and $d(t)$ denote the height of the UAV from the ground and UAV-user distance, respectively. Furthermore, $b1$ and $b2$ are constant values reflecting the environmental impact, and ζ is also a constant value which is determined by using the antenna and the environment parameters. With P_{LoS} being the LoS probability, NLoS probability is given by $P_{NLoS} = 1 - P_{LoS}$. Following the free-space path loss model, the channel's power gain between the UAV and user at time instant t is obtained as

$$g(t) = K_0^{-1} d^{-\alpha}(t) [P_{LoS}\mu_{LoS} + P_{NLoS}\mu_{NLoS}]^{-1} \qquad (2)$$

where, $K_0 = (4\pi f_c/c)^2$, α is the path loss exponent, μ_{LoS} and μ_{NLoS} are the attenuation factors corresponding to LoS and NLoS links, respectively. Further, f_c denotes the carrier frequency and c represents the speed of the light. Finally, the distance from the UAV to the user at time t is defined as

$$d(t) = \sqrt{h^2(t) + [x_{UAV}(t) - x_{user}(t)]^2 + [y_{UAV}(t) - y_{user}(t)]^2} \qquad (3)$$

The received signal-to-noise ratio (SNR) at the user is obtained as

$$\Gamma(t) = \frac{p(t)g(t)}{\sigma^2} \qquad (4)$$

where, $\sigma^2 = BN_0$ with N_0 denoting the power spectral density of the additive white Gaussian noise (AWGN). Further, $p(t)$ and $g(t)$ denote the transmit power of the UAV and channel gain between the UAV and the user. Since we consider only one UAV-user link, we avoid the consideration of interference. The instantaneous rate achieved by the user at time t, $r(t)$, is expressed as

$$r(t) = B \log_2 \left(1 + \frac{p(t)g(t)}{\sigma^2} \right) \qquad (5)$$

C. LINE-OF-SIGHT COMMUNICATION

LoS communication is communication between transceiver pairs with a direct path. The presence of obstacles and terrains in the path causes the diffraction, reflection, and refraction of the signal which reduces the signal strength and makes the path non-LoS. Thus, the LoS paths provide higher throughput as compared to NLoS paths. Thus, one should focus on providing the LoS path for a transceiver pair for better throughput performance [28].

D. PATH REPRESENTATION

The objective of the algorithm is to determine a 3D path comprising multiple nodes for a UAV, starting from a ground location and reaching a destination in the sky. This path planning process takes into account varying heights. To simplify and optimize the path, this research adopts a modeling approach as explained in [7]. For each particle i in the algorithm, the initialization of V_i and X_i are performed, followed by the calculation of the corresponding fitness value f_i. At each step, the algorithm compares the current fitness value f_i with the previous value f_{i-1} and updates it accordingly. When the global position condition is met, a new node denoted as Ψ_g is created, and the algorithm continues. Whenever a new node Ψ_g is established with its corresponding coordinates (x_{n+1}, y_{n+1}), the UAV's height is incrementally increased by a fixed value until it reaches a height h_{n+1}. Consequently, the UAV's new location is updated as $(x_{n+1}, y_{n+1}, h_{n+1})$. This approach effectively simplifies the original 3D path planning problem into a 2D problem. Additionally, the algorithm evaluates the global position conditions and the P_{LoS} condition. If both the global position conditions and the P_{LoS} condition are satisfied, the goal position is updated to become the endpoint of the path.

E. PROBLEM FORMULATION

Before planning the path, one should focus on finding the endpoint. In this work, the endpoint is the point at which the UAV is in LoS with the user so as to provide the required rate. As can be seen from Fig. 1, point B is the optimal end point at which we get the maximum rate at the user. All the traditional path planning algorithms are designed to find the optimum path to the point B which is located directly above the user at a height h. However, it will increase the total distance travelled by the UAV which further increases the energy consumption and travel time. Thus, to balance the energy consumption and rate requirements of the user, we find an optimal region over which the UAV is still in LoS with the user which is obtained by using (1). Let r_s denotes the radius around point B within which the UAV is in LoS with the user and offers the required rate. For example, point D in Fig. 1 is the endpoint that is the nearest point to the source and is within r_s to optimal endpoint B.

From the above discussion, we propose a new fitness function whose objective is to minimize the energy consumption at the UAV which is defined as

$$
\begin{aligned}
f = \arg\min_{P_{LoS}} \quad & E && (6)\\
\text{s.t.} \quad & P_{LoS} \geq \tau && (6a)\\
& d_{ij} < radius(Obs_j) && (6b)\\
& \tau > 0,\ d_{ij} \geq 0 && (6c)
\end{aligned}
$$

where E is given by

$$E = E_{path} + E_{curve} \tag{7}$$

Further, the first constraint in (6a) denotes that the probability of LoS should be greater than a threshold τ and the second constraint defined in (6b) is the condition to avoid the obstacles in the UAV path.

Here, E is the total energy required by the UAV for the entire flight. τ is the minimum threshold for P_{LoS} to satisfy the rate requirement of the user. Here, the UAV path planning has been formulated as a constrained optimization problem with the fitness function containing the travelling distance and risk with three constraints: height, angle, and limited slope of the UAV. The maximum value of τ will be obtained at the optimal endpoint B which is vertically above the user at a height h. E_{path} is the energy consumed for all paths and E_{curve} indicates the extra energy required for the UAV for turning an angle and climbing a height. Among E_{path} and E_{curve}, E_{path} is proportional to the distance flown by the UAV. Thus, the proposed framework is focusing on minimizing E_{path} while maximizing the sum rate and throughput. The expressions of E_{path} and E_{curve}, respectively, are obtained as

$$
\begin{aligned}
&E_{path}\\
&= E_0 \sum_{i=1}^{s-1} \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2 + (z_i - z_{i+1})^2}
\end{aligned}
\tag{8}
$$

and

$$E_{curve} = E_1 \sum_{i=1}^{s-1} \mu_0 * (1 + cos\theta_e) + E_2 \sum_{i=1}^{s-1} |\Delta Z| \tag{9}$$

Here, x_i, y_i, and z_i denote x, y, and z coordinates, respectively, of node i. s denotes the total number of intermediate nodes taken by the UAV to reach the endpoint and E_0 is the energy scaling factor which is defined as the energy required by the UAV to fly a unit distance. μ_0 is the turning angle coefficient and θ represents the angle between two lines which are adjacent to each other [14]. The variation in height between two adjacent nodes is represented by $|\Delta Z|$. The fitness function is performed by cubic spline interpolation and the summation is performed for each slice of the fitness. This is similar to the energy expenditure in the real-time scenario.

IV. PROPOSED PSO BASED APPROACH

Particle Swarm Optimization (PSO) is an approach for finding solutions for problems which can be represented as a point in a S-dimensional solution space. In PSO, the term "particles" refers to the population which is having negligible mass or volume points in the solution space and is subjected to the updations in velocity and position towards a better mode of behaviour.

In path planning, the swarm of particles represents a set of candidate paths from the starting position to the target position while avoiding obstacles. The algorithm iteratively updates the position and velocity of the particles in the swarm based on their current position and velocity. The position of each particle represents a candidate path, and the fitness function evaluates its quality based on factors such as distance and obstacle proximity. The algorithm updates each particle's position and velocity based on the best solution found by itself and the swarm. The swarm gradually explores the search space, improving the quality of paths until an optimal solution is found. The particle with the best fitness value represents the best path from the starting position to the target position while avoiding obstacles. PSO algorithm uses the collective behavior of the swarm to find an optimal path, making it a useful tool for autonomous navigation in robotics, unmanned aerial vehicles, and mobile robots.

This algorithm is initialized by generating a population of random particles $(X_1, X_2, X_3, \cdots, X_S)$ which are uniformly distributed in the search space. In path planning, the swarm of particles represents a set of candidate paths from the starting position to the target position while avoiding obstacles. The position and velocity of the i^{th} particle is represented by $X_i = (x_{i1}, x_{i2}, x_{i3}, \cdots, x_{iS})$ and $V_i = (v_{i1}, v_{i2}, v_{i3}, \cdots, v_{iS})$, respectively. At each iteration, the best position (pbest) of the ith particle and the best position of the population (gbest) is determined as $\Psi_i = (\psi_{i1}, \psi_{i2}, \psi_{i3}, \cdots, \psi_{iS})$ and $\Psi_g = (\psi_{g1}, \psi_{g2}, \psi_{g3}, \cdots, \psi_{gS})$, respectively. The new velocity and position of each particle are then updated as

$$V_i^{k+1} = wV_i^k(t) + c_1r_1(\Psi_i - X_i^k) + c_2r_2(\Psi_g - X_i^k) \tag{10}$$

and

$$X_i^{k+1} = \Psi_i^k + V_i^{k+1} \tag{11}$$

respectively. Here, $i = 1, 2, \cdots, M$ and $k = 1, 2, \cdots, K$ where, M denotes the population size and K is the iteration number. Inertia weight is represented by w which takes either a constant or a dynamically changing value [30]. This parameter regulates the exploration capacity of the search space. Hence, it is recommended to set a higher value initially such as 0.9 so that it allows the particles to move faster in the direction of the global optimum. After achieving the global optimum, the values can be decreased to 0.4 to limit the search capacity of the search space. Thus, the algorithm shifts from an exploratory mode to an exploitative mode. c_1 and c_2 are the learning factors whose value is 2. r_1 and r_2 denote the random numbers generated uniformly at random from 0 to 1. The velocity and position of the particle are initialised based on the following constraints

$$-v_{\max} \leq v_{id} \leq v_{\max} \tag{12}$$

and

$$x_{\min} \leq x_{id} \leq x_{\max} \tag{13}$$

where, $v_{\max}$ is the maximum allowed velocity that a particle can attain to keep the maximum global exploration ability of PSO in control. The inertia weight w maintains the balance between the global and local exploration abilities of PSO. This parameter decides the converging properties of the PSO in terms of optimality and speed. $x_{\min}$ and $x_{\max}$ are the lower and upper boundary values of the solution space. A problem-dependent predefined fitness function is used to calculate the performance of each particle in the solution space. At each iteration, every particle tries to attain fitness and moves towards the direction which satisfies the fitness function.

TABLE 2: Comparison of P_{LoS} and achievable rate at various UAV locations.

User Location	UAV Location	UAV-User distance	P_{LoS}	Rate (Mbps)	% Rate
(1950,1950,0)	(1000,1000,800)	1563.6495	0.6424	24.26	83.84%
(1950,1950,0)	(1100,1100,800)	1443.9529	0.6654	24.05	85.49%
(1950,1950,0)	(1200,1200,800)	1328.5330	0.6891	24.29	87.23%
(1950,1950,0)	(1300,1300,800)	1218.6057	0.7137	24.54	89.06%
(1950,1950,0)	(1400,1400,800)	1115.7956	0.7394	24.80	90.96%
(1950,1950,0)	(1500,1500,800)	1022.2524	0.7662	25.05	92.9%
(1950,1950,0)	(1600,1600,800)	940.7443	0.7940	25.29	94.84%
(1950,1950,0)	(1700,1700,800)	874.6427	0.8225	25.50	96.68%
(1950,1950,0)	(1800,1800,800)	827.6472	0.8510	25.66	98.3%
(1950,1950,0)	(1900,1900,800)	803.1189	0.8784	25.75	99.54%
(1950,1950,0)	(1950,1950,800)	800.0	0.8913	25.76	100%

Algorithm 1: Proposed PSO algorithm for UAV path planning in Scenario I.

```
Input: $v_{max}, v_{min}$, goal, origin, solution space
Output: Optimized path for UAV
Initialize $M, S, K, w, \tau$;
for particles do
    Initialize $V_i$ and $X_i$ of particles;
    Calculate fitness $f_i$;
    Set $\Psi_i = f_i$;
end
Set $\Psi_g = f_i$;
for Iterations do
    for Particle i do
        Update $V_i$ and $X_i$ ;
        Calculate fitness $f_i$;
        if $f_i < \Psi_{i-1}$ then
            $\Psi_i = f_i$;
        end
    end
    if $\Psi_i < \Psi_g$ and $P_{LoS} \geq \tau$ then
        $\Psi_g = \Psi_i$;
        Set goal$_{new}$= $\Psi_g$;
        break;
    end
    if $\Psi_i < \Psi_g$ then
        $\Psi_g = \Psi_i$;
        node$_{new}$= $\Psi_g$;
    end
end
Return $\Psi_g$
```

A. PROPOSED MODIFIED PARTICLE SWARM OPTIMIZATION ALGORITHM

The objective of the proposed PSO algorithm is to estimate the nearest endpoint at which the user is in LoS with UAV so as to satisfy the required rate. The LoS probability, P_{LoS}, defined in (1) is used to check the LoS communication link. Table 2 presents the achievable rate at different locations of the UAV. It is observed from Table 2 that the deviation of the achievable rate is 3% from the maximum when P_{LoS} is 0.83. Thus, the proposed PSO algorithm checks after each particle update whether the required rate is achieved. In path planning, the fitness function of normal PSO typically considers factors such as distance to the target position, proximity to obstacles, and smoothness of the path. When PSO iterates to find the best position for a node and the point

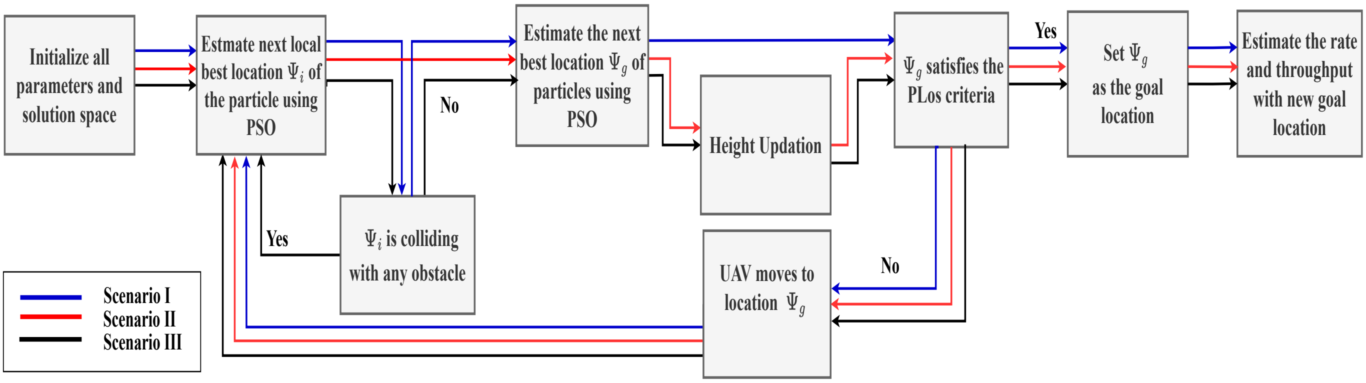

Initialize all parameters and solution space
Estmate next local best location Ψ_i of the particle using PSO
No
Estimate the next best location Ψ_g of particles using PSO
Height Updation
Ψ_g satisfies the PLos criteria
Yes
Set Ψ_g as the goal location
Estimate the rate and throughput with new goal location
Yes
Ψ_i is colliding with any obstacle
No
UAV moves to location Ψ_g
Scenario I
Scenario II
Scenario III

FIGURE 3: Block diagram of the proposed path planning.

Algorithm 2: Proposed PSO algorithm for UAV path planning in Scenario II.

```
Input: $v_{max}, v_{min}$, goal, origin, solution space
Output: Optimized path for UAV
Initialize $M, S, K, w, \tau$, $h_t$=0;
for particles do
    Initialize $V_i$ and $X_i$ of particles;
    Calculate fitness $f_i$;
    Set $\Psi_i = f_i$;
end
Set $\Psi_g = f_i$;
Set $Z_{eq}$=0;
for Iterations do
    for Particle $i$ do
        Update $V_i$ and $X_i$ ;
        Calculate fitness $f_i$;
        if $f_i < \Psi_{i-1}$ then
            $\Psi_i = f_i$;
        end
    end
    if $\Psi_i < \Psi_g$ and $P_{LoS} \geq \tau$ then
        $\Psi_g = \Psi_i$;
        Set goal$_{new}$= $\Psi_g$;
        break;
    end
    if $\Psi_i < \Psi_g$ then
        $\Psi_g = \Psi_i$;
        node$_{new}$= $\Psi_g$;
        Calculate $Z_{eq}$
    end
    $h_t$.append= $Z_{eq}$
end
Return goal$_{new}$ and $h_t$;
```

Ψ_g satisfies the P_{LoS} criteria, the algorithm sets Ψ_g as the new endpoint for UAV and evaluates the rate of the user from this location. Usually, the closing criterion of PSO algorithm is the target fitness value. In path planning, the fitness value can represent the distance between the optimal path and the target position or the number of obstacles avoided. Here, in this proposed approach, we check whether the corresponding P_{LoS} satisfies the criteria of less than 3 % rate loss at each updation of Ψ_g. This considers the closing criteria of our modified PSO algorithm. To analyse the robustness of the proposed work, we tested the algorithm with three different scenarios. Fig. 3 shows the block diagram of the proposed PSO algorithm for all the scenarios. A detailed description of these three scenarios is given in the following section.

Scenario I: In this scenario, we consider that the UAV is already at a height of h at the start point and the proposed algorithm needs to find the optimal path to the endpoint which is considered to be at the same height. Since the altitude of the UAV is constant, the path-planning environment becomes a 2D space. Here, we consider 12 different obstacles present on the plane at different heights greater than h. Since the UAV is already at height h, the algorithm only deals with (x, y) coordinates of the obstacles in the vicinity of the UAV. Hence, for choosing the next best position of the swarm Ψ_g, the algorithm checks whether the point is inside any obstacles so as to change the path. The algorithm directs the UAV until the new location Ψ_g satisfies the P_{LoS} condition. Once the condition is satisfied, the algorithm sets Ψ_g as the endpoint at which there is LoS between UAV and the user. Algo. 1 describes the details of the proposed algorithm for the scenario I.

Scenario II: The second scenario is a 3D environment where the initial position of the UAV is assumed to be on the ground. Hence, in this case, the algorithm tries to plan a 3D path for the UAV from the ground (start point) to the endpoint which is in the sky. As this involves 3D path planning, the height also varies during path planning. To simplify the process and make the path a smooth one, this work adopts the modelling approach defined in [7]. Each time a position is decided, the algorithm increases the height of the UAV by a fixed amount. This approach will still simplify the 3D path planning problem into a 2D one. The pseudo-code for this approach is given in Algo. 2. For each particle i, the algorithm will initialize V_i and X_i and calculates the fitness value f_i. At each point, the algorithm checks whether the f_i is

Algorithm 3: Proposed PSO algorithm for UAV path planning in Scenario III.

```
Input: $v_{max}, v_{min}$, goal, origin, Solution Space
Output: Optimized Path for UAV
Initialize $M, S, K, w, \tau$;
for particles do
    Initialize $V_i$ and $X_i$ of particles;
    Calculate Fitness $f_i$;
    Set $\Psi_i = f_i$;
end
Set $\Psi_g = f_i$;
Set $Z_{eq}$=0;
for Iterations do
    for Particle $i$ do
        Update $V_i$ and $X_i$ ;
        Calculate the fitness $f_i$;
        for Obstacles $j$ do
            Calculate the distance $d_{ij}$ between $f_i$ and
             $Obs_j$ center;
            if $d_{ij} < radius(Obs_j)$ then
                break;
            end
        end
        if $f_i < \Psi_{i-1}$ then
            $\Psi_i = f_i$;
        end
    end
    if $\Psi_i < \Psi_g$ and $P_{LoS} \geq \tau$ then
        $\Psi_g = \Psi_i$;
        Set goal$_{new}$= $\Psi_g$;
        break;
    end
    if $\Psi_i < \Psi_g$ then
        $\Psi_g = \Psi_i$;
        node$_{new}$= $\Psi_g$;
        Calculate $Z_{eq}$
    end
    $h_t$.append= $Z_{eq}$
end
Return goal$_{new}$ and $h_t$;
```

less than the previous fitness f_{i-1} and updates f_i accordingly. It also checks for global position conditions along with the P_{LoS} condition. If the global position condition is satisfied, Ψ_g becomes a new node and the algorithm continues. If both global position conditions along with the P_{LoS} conditions are satisfied, the algorithm will update the goal position as the endpoint.

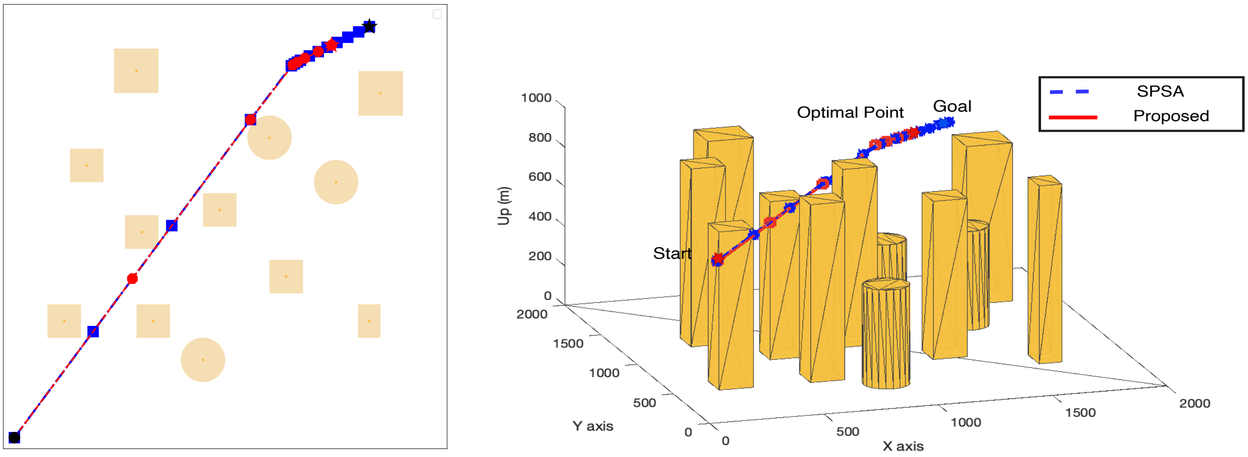
Scenario III: The third scenario is similar to Scenario II, however, we consider the presence of obstacles in the path. The proposed approach tries to detect the obstacle at each step to avoid collision for the UAV. The pseudo-code for this approach is given in Algo. 3. At each iteration, while estimating the local best location (Ψ_i) of the particles, the algorithm checks whether there is an obstacle present at that location. This is done by calculating d_k and R_k from Fig. 2b. If d_k is less than R_k, the proposed algorithm estimates the possibility of a collision at the next location. In order to avoid the collision, every Ψ_i that is located inside the obstacles is discarded and avoided being updated as the global best position even if it satisfies all the required conditions.

TABLE 3: Simulation Parameters [14].

Parameter	**Description**	**Value**
c_1,c_2	learning factors	2
w	Inertia weight	0.72
fc	carrier frequency	2GHz
N_0	Noise Power Spectral	-170 dBm/Hz
BW	Bandwidth	1MHz
$b1$	Environmental Parameter	0.36 [27]
$b2$	Environmental Parameter	0.21 [27]
α	Path loss exponent	2
μ_{LoS}	Additional Path loss for LoS	3dB [27]
μ_{NLoS}	Additional Path loss for NLoS	23dB [27]
v_{max}	Maximum allowed velocity of a particle	1000
$x_{\min}$	Lower boundary value of the solution space	(0, 0)
$x_{\max}$	Upper boundary value of the solution space	(2000, 2000)
ν	Speed of the UAV	46 m/s
λ	Amount of energy consumed for UAV to fly over a unit distance	0.7 Joules

TABLE 4: Locations and radii of all the obstacles considered for all the scenarios.

Object type	**Location**	**Radius**	**Height**		
			Scenario I	**Scenario II**	**Scenario III**
Square Cylinder	(275,575,0)	150	800	100	600
Square Cylinder	(675,575,0)	15	900	400	600
Square Cylinder	(375,1275,0)	150	900	300	500
Square Cylinder	(625,975,0)	150	800	300	500
Square Cylinder	(1650,575,0)	150	900	300	300
Square Cylinder	(1275,775,0)	150	800	150	300
Square Cylinder	(600,1700,0)	200	900	300	300
Square Cylinder	(1700,1600,0)	200	800	500	1700
Square Cylinder	(975,1075,0)	150	900	300	1150
Circular Cylinder	(1500,1200,0)	100	500	500	500
Circular Cylinder	(900,400,0)	100	400	250	500
Circular Cylinder	(1200 1400,0)	10	400	400	400



(a) 2D illustration

(b) 3D illustration

FIGURE 4: An illustration of the optimal path obtained by the proposed approach and SPSA in scenario I. Here, circle and square markers on the plot represent the steps the UAV takes.

TABLE 5: Performance evaluation of the proposed approach and SPSA in terms of energy consumption, distance travelled, travel time, and rate.

Parameter	**Scenario I**		**Scenario II**		**Scenario III**		
	SPSA [14]	**Proposed**	**SPSA [14]**	**Proposed**	**SPSA [14]**	**Proposed with end point B**	**Proposed with end point D**
Energy (J)	2065.91	1881.78	2012	1419	1833.16	1849.80	1649.94
Distance (m)	2758.22	2512.39	2687	1895	2447.47	2469.69	2202.8
Time (s)	59.96	54.61	58.41	41.19	53.20	53.68	47.88
Rate (Mbps)	25.63	25.15	24.77	23.57	25.52	25.61	25.09

V. PERFORMANCE EVALUATION

In this section, we provide the experimental results corresponding to the proposed framework for all the scenarios.

A. EXPERIMENTAL CONDITIONS

For all scenarios, we consider a geographical area of $2000 \times 2000\ m^2$. For scenario I, we consider $(50, 50, 800)$ as the start point of the UAV and the user is placed at $(1950, 1950, 800)$. The experimental conditions for scenario II are similar to the conditions defined in [14]. In this scenario, the start point of the UAV is $(50, 50, 0)$ and the user is located at $(1950, 1950, 0)$. The environmental conditions for scenario III are similar to scenario II except that the user is placed at $(1650, 1900, 0)$. Table 3 list the all parameter values considered for experimental results. Table 4 provides the locations of the obstacles and their heights for all the scenarios.

B. PERFORMANCE MEASURES

1) Energy Consumption

With λ being the amount of energy consumed for UAV fly over a unit distance, the total energy consumed for travelling a distance d is obtained as

$$E_{path} = \lambda \cdot d \tag{14}$$

2) Travel Time

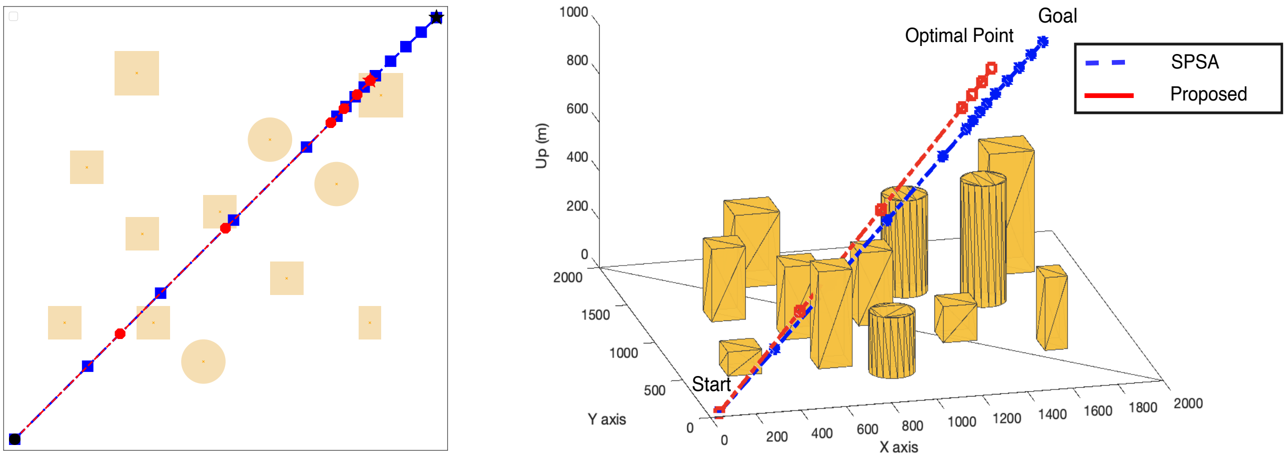
The total time taken for travelling a distance d with a speed ν is obtained as

$$\tau = \frac{d}{\nu} \tag{15}$$

C. EXPERIMENTAL RESULTS

The optimum path obtained using the proposed approach and SPSA in scenario I is demonstrated in Fig. 4. Fig. 4a illustrates the 2D view of the path and Fig. 4b demonstrates the 3D path of the UAV. It is clear from Fig. 4b that the proposed framework reduces the total path travelled by the UAV without much compromising on the achievable rate at the user. This results in reduced energy consumption, distance travelled, and the travel time. Table 5 shows the total energy consumption, distance travelled, and the time taken for the travel for all the algorithms. It is observed from Table 5 that the total energy consumed, distance travelled, and time taken by the UAV, with the proposed framework, are 1881.78 Joules (J), 2512.39 meters (m), and 54.61 seconds (s), respectively. Further, the total energy consumed, distance travelled, and time taken by the UAV, with SPSA, are 2065.91 J, 2758.22 m, and 59.96 s, respectively. Thus, it can be concluded that the proposed framework outperforms SPSA in scenario I.

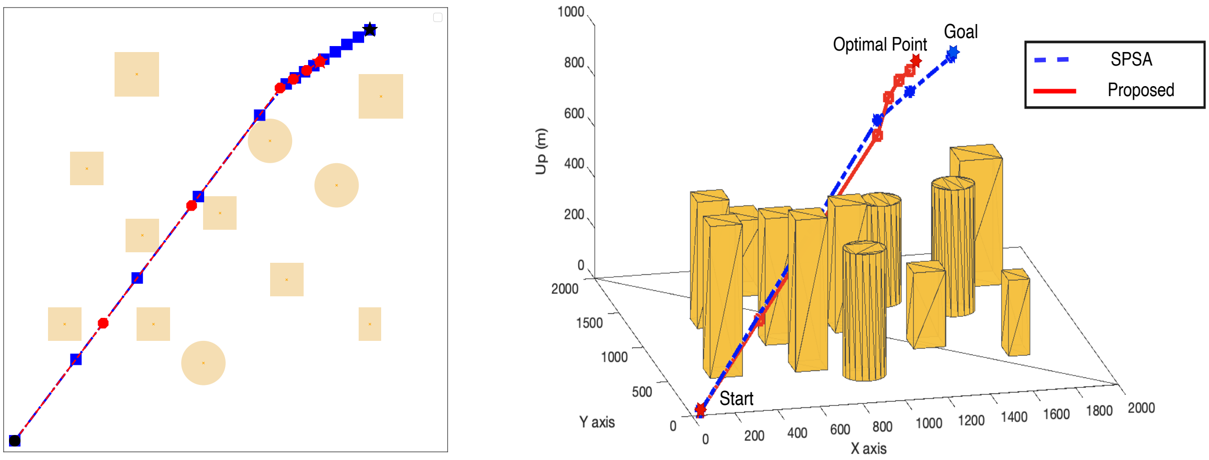
The optimum path obtained using the proposed approach



(a) 2D illustration

(b) 3D illustration

FIGURE 5: An illustration of the optimal path obtained by the proposed approach and SPSA in scenario II. Here, circle and square markers on the plot represent the steps the UAV takes.



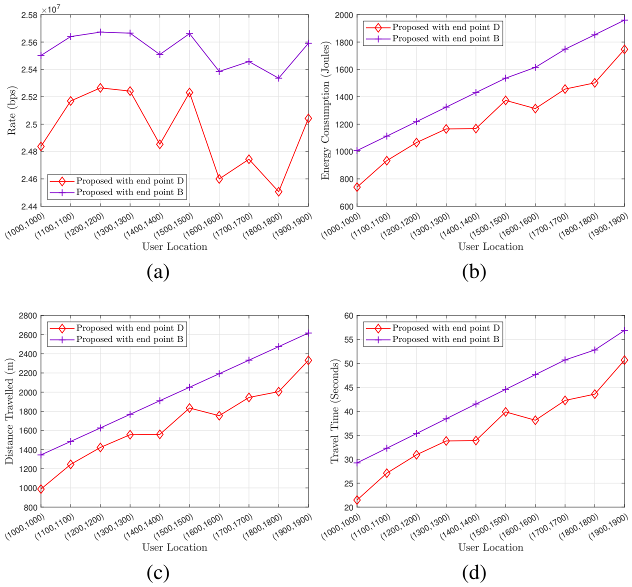
(a) 2D illustration

(b) 3D illustration

FIGURE 6: An illustration of the optimal path obtained by the proposed approach and SPSA in scenario III. Here, circle and square markers on the plot represent the steps the UAV takes.

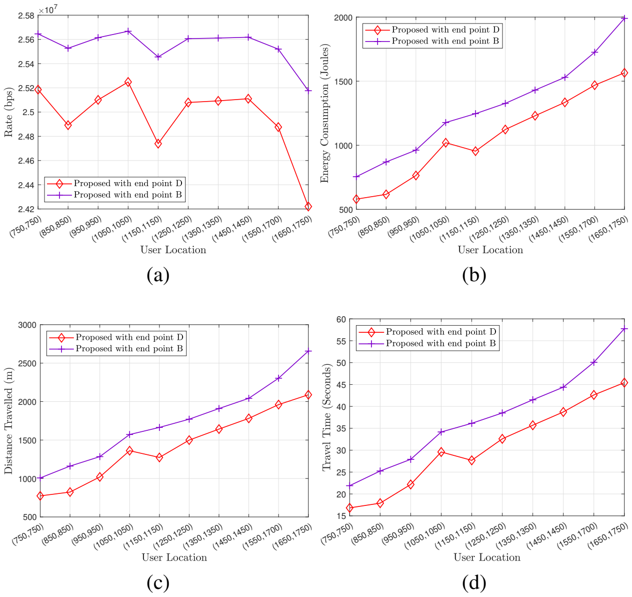
in scenario II is demonstrated in Fig. 5. Fig. 5a illustrates the 2D view of the path and Fig. 5b demonstrates the 3D path of the UAV. It is clear from Fig. 5b that the proposed framework reduces the total path travelled by the UAV without much compromising on the achievable rate at the user. This results in reduced energy consumption, distance travelled, and the travel time. Table 5 shows the total energy consumption, distance travelled, and the time taken for the travel. It is observed from Table 5 that the total energy consumed, distance travelled, and time taken by the UAV, with the proposed framework, are 1419 J, 1895 m, and 41.19 s, respectively. Further, the total energy consumed, distance travelled, and time taken by the UAV, with SPSA, are 2012 J, 2687 m, and 58.41 s, respectively. Thus, it can be concluded that the proposed framework outperforms the SPSA in scenario II.

The optimum path obtained using the proposed approach in scenario III is demonstrated in Fig. 6. Fig. 6a illustrates the 2D view of the path followed and Fig. 6b demonstrates the 3D path followed by the UAV. It is clear from Fig. 6b that the proposed framework reduces the total path travelled by the UAV without much compromising the achievable rate at the user. This results in reduced energy consumption, distance travelled, and the delay. Table 5 shows the total energy consumption, distance travelled, and the time taken for the travel. It is observed from Table 5 that the total energy consumed, distance travelled, and time taken by the UAV, with the proposed framework are 1649.94 J, 2202.8 m, and 47.88 s, respectively. Further, the total energy consumed, distance travelled, and time taken by the UAV, with the conventional approach, are 2012 J, 2687 m, and 58.41 s,



(a) (b) (c) (d)

FIGURE 7: An illustration of (a) sum rate, (b) energy consumption, (c) distance saved, and (d) time saved at different locations of the user in Scenario II.



(a) (b) (c) (d)

FIGURE 8: An illustration of (a) sum rate, (b) energy consumption, (c) distance saved, and (d) time saved at different locations of the user in Scenario III.

respectively. Thus, it can be concluded that the proposed framework outperforms the conventional one for scenario III.

Figs. 7a, 7b, 7c, and 7d show the variation of sum rate, energy consumption, distance travelled, and time of travel with respect to different locations of user for the proposed algorithm with end point chosen to be D and B in scenario II. It is observed from Figs. 7b, 7c, and 7d that by choosing the destination point as D the energy consumption, distance travelled, and time of travel are less in comparison to when we choose the destination point as B. However, the sum rate achieved are comparable when choosing both the destinations.

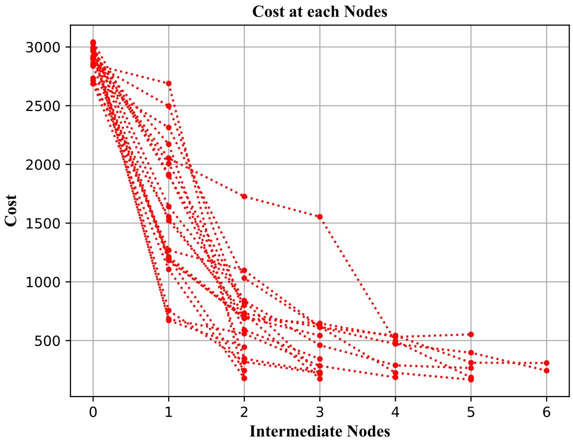


FIGURE 9: The comparison of the cost obtained at each node of the optimum path for different iterations in scenario II.

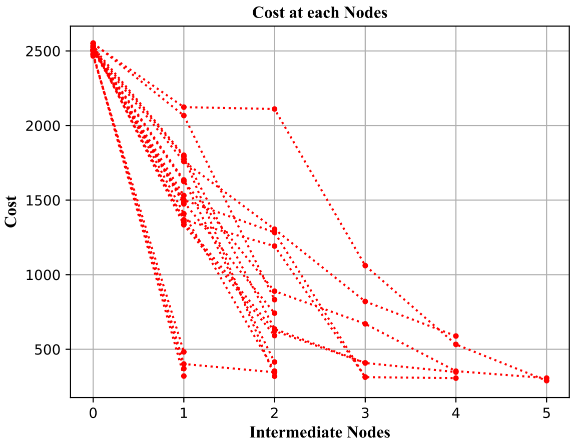
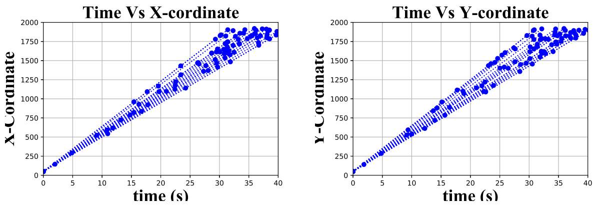


FIGURE 10: The comparison of the cost obtained at each node of the optimum path for different iterations in scenario III.

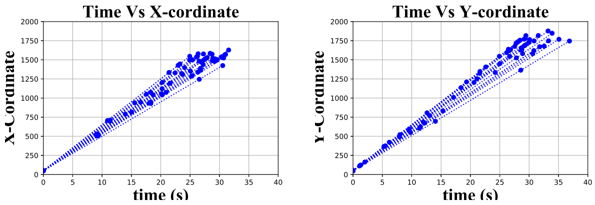
Figs. 8a, 8b, 8c, and 8d show the variation of sum rate, energy consumption, distance travelled, and time of travel with respect to different locations of user for proposed algorithm with end point chosen to be D and B in scenario III. It is observed from Figs. 8b, 8c, and 8d that by choosing the end point as D the energy consumption, distance travelled, and time of travel are less in comparison to when we choose the end point as B. However, the sum rate achieved are comparable when choosing both the destinations.

Figs. 9 and 10 show the distribution of cost for different paths followed during different iterations in scenario II and scenario III, respectively. The cost value is estimated by



(a) (b)

FIGURE 11: Comparison of X &Y -coordinate Vs Time for various UAV velocities for scenario II.



(a) (b)

FIGURE 12: Comparison of X &Y -coordinate Vs Time for various UAV velocities for scenario III.

TABLE 6: The performance comparison of the proposed algorithm with existing ones in terms of mean, worst, optimal, and standard deviation.

Parameter	CSO [14]	DE [14]	GWO [14]	PSO [14]	SSA [14]	SPSA [14]	Proposed
Mean	6221.22	4562.37	3958.70	4683.83	2784.55	2004.95	1918.37
Worst	6702.69	6625.81	6902.46	5182.99	3081.59	2024.60	1986.173
Optimal	5251.67	3473.63	3294.27	4143.35	2525.77	1999.51	1895.173
Standard	381.40	752.85	578.59	297.68	183.07	5.84	13.70

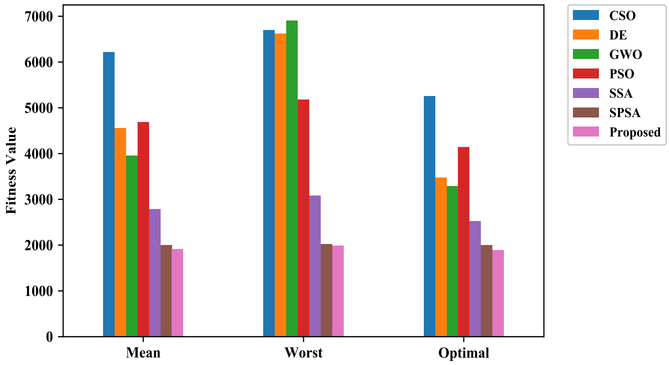


FIGURE 13: The performance comparison of the proposed algorithm with existing ones in terms of fitness values.

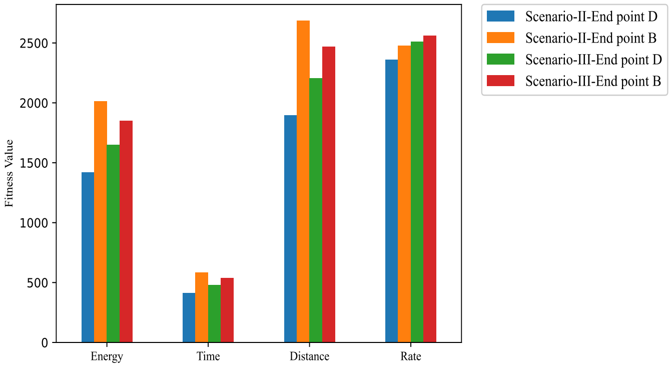


FIGURE 14: Performance evaluation of the proposed algorithm for different scenarios in terms of fitness value.

calculating the distance between the coordinates of a path node and the target point. The selection of the next path node is also depends on this cost criteria. The selection criteria for each path node (particle) in PSO algorithm are based on the concept of fitness evaluation. The fitness function is defined based on the specific problem or objective being optimized. Here, we utilizes the LoS probability to find an optimal destination location where the UAV is in LoS with the user and can provide the required downlink rate. The proposed modified PSO algorithm checks whether the corresponding PLoS satisfies the criteria of less than 3 % rate loss at each updation of Ψ_g where, Ψ_g is the new path node for UAV. This considers the cost criteria of our modified PSO algorithm. If the cost at a particular node is larger than its previous one, it is assumed that the particular node location is not estimated in an optimal way. By proceeding from one path node to the next, the algorithm is always tries to converge with minimum cost.

In Figs. 11 and 12, the variation of path of UAV in x and y direction with respect to time is depicted for scenario II and III, respectively. As the proposed approach tries to estimate the minimum distance path, the path nodes will be always in straight line with respect to time for a constant velocity. Hence, we tried to analyse the nature of selection of nodes by the proposed approach for different random velocities in a predefined range.

The mean, worst, optimal, and standard deviation values of the cost function, averaged over 1000 iterations for different algorithms, are given in Table 6. It is observed from Table 6 that CSO algorithm has the worst convergence effect for the given scenario in path planning. It is also noticed that as the number of iteration increases, the convergence effect is getting better for DE, PSO, and SSA algorithms. Finally, it is shown that the proposed approach is outperforming other algorithms in terms of lowest cost value due to the optimal conditions we applied in the proposed PSO approach. The corresponding graph is given in Fig. 13. Further, the variation of fitness value for the proposed algorithm with end points B and D is depicted in Fig. 14 for scenarios II and III. Effect of obstacle avoidance approach on the optimum value of various attributes such as energy, distance, time, and rate are plotted and compared. In the end, it is obvious that the obstacle avoidance part is adding some distance to the planned path of UAV in order to avoid collisions. For a better illustration, time and rate attributes are scaled up and scaled down, respectively, and plotted in Fig. 14.

VI. CONCLUSION AND FUTURE WORK

In this work, a novel framework has been proposed that utilizes the modified Particle Swarm Optimization (PSO) algorithm for path planning of UAV in order to deliver the data to a user with required rate. Here, a joint problem has been formulated in terms of path planning and the energy consumption in order to improve the instantaneous sum rate of the user. Further, the line of sight probability has been incorporated to satisfy the rate requirements of the user. It has been shown through simulations that the proposed framework reduces the energy consumption, distance travelled, and time of travel in addition to providing the required rate when compared to other methods in three different scenarios. We have also shown that the fitness value is less for the proposed work when compared to other works. In future, we plan to evaluate the performance for multi-UAV and multi-user system. Further, we will explore the effectiveness and efficiency of PSO algorithm in rural areas.

REFERENCES

[1] Z. Wu, J. Li, J. Zuo, and S. Li, “Path planning of UAVs based on collision probability and Kalman filter,” *IEEE Access*, vol. 6, pp. 34237–34245, Jul. 2018.

[2] A. Mardani, M. Chiaberge, and P. Giaccone, “Communication-aware UAV path planning,” *IEEE Access*, vol. 7, pp. 52609–52621, Apr. 2019.

[3] J. Tisdale, Z. Kim, and J. K. Hedrick, “Autonomous UAV path planning and estimation,” *IEEE Robotics & Automation Magazine*, vol. 16, no. 2, pp. 35–42, Jun. 2009.

[4] R. Xie, Z. Meng, L. Wang, H. Li, K. Wang, and Z. Wu, “Unmanned aerial vehicle path planning algorithm based on deep reinforcement learning in large-scale and dynamic environments,” *IEEE Access*, vol. 9, pp. 24884–24900, Feb. 2021.

[5] P.-C. Song, J.-S. Pan, and S.-C. Chu, “A parallel compact cuckoo search algorithm for three-dimensional path planning,” *Applied Soft Computing*, vol. 94, pp. 106443, Sep. 2020.

[6] V. Roberge, M. Tarbouchi, and G. Labonté, “Fast genetic algorithm path planner for fixed-wing military UAV using GPU,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 5, pp. 2105–2117, Oct. 2018

[7] Y. Chen, J. Yu, Y. Mei, Y. Wang, and X. Su, “Modified central force optimization (MCFO) algorithm for 3D UAV path planning,” *Neurocomputing*, vol. 171, pp. 878-–888, Jan. 2016.

[8] C. Xu, H. Duan, and F. Liu, “Chaotic artificial bee colony approach to uninhabited combat air vehicle (UCAV) path planning,” *Aerospace science and technology*, vol 18, no. 4, pp. 535–541, Dec. 2010.

[9] X. Yu, W. -N. Chen, T. Gu, H. Yuan, H. Zhang, and J. Zhang, “ACO-A*: Ant colony optimization plus A* for 3-d traveling in environments with dense obstacles,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 4, pp. 617–631, Aug. 2019.

[10] Y. V. Pehlivanoglu and P. Pehlivanoglu, “An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems,” *Applied Soft Computing.*, vol. 112, p. 107796, 2021.

[11] R. Eberhart, Y. Shi, and J. Kennedy, “Swarm intelligence,” *Elsevier*, Apr. 2001.

[12] Z.-L. Gaing, “Particle swarm optimization to solving the economic dispatch considering the generator constraints,” *IEEE Transactions on Power Systems,*, vol. 18, no. 3, pp. 1187–1195, Aug. 2003.

[13] R.C. Eberhart and Y. Shi, “Comparison between genetic algorithms and particle swarm optimization,” *Lecture Notes in Computer Science Evolutionary Programming VII*, vol. 1447, pp. 611–616, Mar. 1998.

[14] W. Yu, J. Liu, and J. Zhou, “A novel sparrow particle swarm algorithm (SPSA) for unmanned aerial vehicle path planning,” *Scientific Programming 2021*, pp. 5158304, Dec. 2021.

[15] L. Yang, J. Qi, J. Xiao, and X. Yong, “A literature review of UAV 3D path planning,” in *Proc. of the 11th World Congress on Intelligent Control and Automation*, Shenyang, Jun. 2014, pp. 2376–2381.

[16] Y. Zhao, Z. Zheng, and Y. Liu, “Survey on computational-intelligence-based UAV path planning,” *Knowledge-Based Systems*, vol. 158, pp.54–64, Oct. 2018.

[17] Z. Qadir, F. Ullah, H.S. Munawar, and F. Al-Turjman, “Addressing disasters in smart cities through UAVs path planning and 5G communications: A systematic review,” *Computer Communications*, vol. 168, pp. 114–135, Feb. 2021.

[18] Y. V. Pehlivanoglu, “A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV,” *Aerospace Science and Technology*, vol. 16, no. 1, pp. 47–55, Jan. 2012.

[19] M. D. Phung and Q. P. Ha, “Safety-enhanced UAV path planning with spherical vector-based particle swarm optimization,” *Applied Soft Computing*, vol. 107, pp. 107376, Aug. 2021.

[20] X. Ou, Y. Liu and Y. Zhao, “PSO based UAV online path planning algorithms,” *Proceedings of International Conference on Automation, Control and Robots (ICACR)*, pp. 41-45, Dec. 2017.

[21] C. Huang and J. Fei,“UAV path planning based on particle swarm optimization with global best path competition,” *International Journal of Pattern Recognition And Artificial Intelligence*, vol. 32, no. 6, Jun. 2018.

[22] J.-J. Shin and H. Bang, “UAV path planning under dynamic threats using an improved PSO algorithm, ” *International Journal of Aerospace Engineering*, vol. 2020, pp. 1-17, Dec. 2020.

[23] S. Shao, Y. Peng, C. He and Y. Du, “Efficient path planning for UAV formation via comprehensively improved particle swarm optimization,” *ISA Transactions*, vol. 97, pp. 415-430, Feb. 2020.

[24] I. Rahman, P. M. Vasant, B. S. M. Singh and M. Abdullah-Al-Wadud, “On the performance of accelerated particle swarm optimization for charging plug-in hybrid electric vehicles,” *Alexandria Eng. J.*, vol. 55, no. 1, pp. 419-426, Mar. 2016.

[25] M. Li, W. Du and F. Nian, “An adaptive particle swarm optimization algorithm based on directed weighted complex network,” *Math. Problems Eng.*, vol. 2014, Apr. 2014.

[26] D. P. Rini, S. M. Shamsuddin and S. S. Yuhaniz, “Particle swarm optimization: Technique system and challenges,” *International Journal of Computer Applications (IJCA)*, vol. 14, no. 1, pp. 19-26, 2011.

[27] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, “Trajectory design and power control for multi-UAV assisted wireless networks: A machine learning approach,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7957–7969, Aug. 2019.

[28] L. De Floriani, P. Marzano, E. Puppo, “Line-of-sight communication on terrain models,” *International journal of geographical information systems*, vol. 8, no. 4, pp. 329–42, Jul. 1994.

[29] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez, and R. G. Harley, “Particle swarm optimization: Basic concepts, variants and applications in power systems,” *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 2, pp. 171–195, Apr. 2008.

[30] Eberhart and Yuhui Shi, “Particle swarm optimization: Developments, applications and resources,” in *Proc. the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, Seoul, Korea (South), May 2001, pp. 81–86.

AMALA SONNY received the B.Tech. degree in electronics and communication engineering from the TKM college of engineering, Kollam, India, in 2014, and the M.Tech. degree from the National Institute of Technology Calicut, Calicut, India, in 2017. She is currently pursuing the Ph.D. degree with the Department of EE, Indian Institute of Technology Hyderabad, Hyderabad, India, . Her research interests include Indoor and outdoor localization, ML-based human and object detection and wireless communication.

SREENIVASA REDDY YEDURI (MEMBER, IEEE) received received the B.E. degree in electronics and communication engineering from Andhra University, Visakhapatnam-India, in 2013, the M.Tech. degree from the ABV-Indian Institute of Information Technology, Gwalior-India, in 2016, and the Ph.D. degree with the Department of Electronics and Communication Engineering, National Institute of Technology, Goa-India, in 2021. He is currently working as a post-doctoral research fellow with autonomous and cyber-physical systems (ACPS) research group, Department of ICT, at University of Agder, Grimstad, Norway. His research interests are AI/ML in Communication, Cyber Physical Systems, Machine-type communications, Internet of Things, LTE MAC, 5G MAC, optimization in communication, wireless networks, power line communications, visible light communications, hybrid communication systems, spectrum cartography, spectrum sensing, UAV-assisted wireless networks, wireless sensor networks, UAV path planning.

LINGA REDDY CENKERAMADDI (SENIOR MEMBER, IEEE) received the master's degree in electrical engineering from the Indian Institute of Technology Delhi (IIT Delhi), New Delhi, India, in 2004, and the Ph.D. degree in electrical engineering from the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, in 2011. He worked at Texas Instruments on mixed-signal circuit design before joining the Ph.D. Program at NTNU. After finishing his Ph.D. degree, he worked on radiation imaging for an atmosphere–space interaction monitor (ASIM mission to the International Space Station) at the University of Bergen, Bergen, Norway, from 2010 to 2012. He is currently the Leader of the Autonomous and Cyber-Physical Systems (ACPS) Research Group and a Professor with the University of Agder, Grimstad, Norway. He has coauthored over 130 research publications that have been published in prestigious international journals and standard conferences in the research areas of Internet of Things (IoT), Cyber-Physical systems, Autonomous systems, Robotics and Automation involving advanced sensor systems, Computer vision, Thermal imaging, LiDAR imaging, Radar imaging, wireless sensor networks, smart electronic systems, advanced machine learning techniques, connected autonomous systems including drones/unmanned aerial vehicles (UAVs), unmanned ground vehicles (UGVs), unmanned underwater systems (UUSs), 5G- (and beyond) enabled autonomous vehicles, and Socio-Technical Systems like urban transportation systems, Smart Agriculture and Smart Cities. He is also quite active in medical imaging. Several of his master students won the best master thesis award in information and communication technology (ICT). Dr. Cenkeramaddi is also a senior member of IEEE, member of ACM and a member of the editorial boards of various international journals and the technical program committees of several IEEE conferences. He serves as a reviewer for several reputed international conferences and IEEE journals. He is the Principal Investigator and a Co-Principal Investigator of many research grants from the Norwegian Research Council.

• • •