Proceedings Paper:

Goudarzi, S., Wang, W., Xiao, P. et al. (2 more authors) (2022) UAV-enabled Edge Computing for optimal task distribution in target tracking. In: Proceedings of the 2022 25th International Conference on Information Fusion (FUSION). 2022 25th International Conference on Information Fusion (FUSION), 04-07 Jul 2022, Linköping, Sweden. Institute of Electrical and Electronics Engineers . ISBN 9781665489416

https://doi.org/10.23919/FUSION49751.2022.9841357

UAV-enabled Edge Computing for Optimal Task Distribution in Target Tracking

Shidrokh Goudarzi[1], Wenwu Wang[1], Pei Xiao[2], Lyudmila Mihaylova[3], and Simon Godsill[4]
[1]Centre for Vision Speech and Signal Processing (CVSSP), The University of Surrey, UK
[2]Institute for Communication Systems (ICS), The University of Surrey, UK
[3]Department of Automatic Control and Systems Engineering, The University of Sheffield, UK
[4]Department of Engineering, University of Cambridge, UK
Email: {[s.goudarzi, w.wang, p.xiao]@surrey.ac.uk, l.s.mihaylova@sheffield.ac.uk, sjg@eng.cam.ac.uk}

***Abstract*—Unmanned aerial vehicles (UAVs) are useful devices due to their great manoeuvrability for long-range outdoor target tracking. However, these tracking tasks can lead to sub-optimal performance due to high computation requirements and power constraints. To cope with these challenges, we design a UAV-based target tracking algorithm where computationally intensive tasks are offloaded to Edge Computing (EC) servers. We perform joint optimization by considering the trade-off between transmission energy consumption and execution time to determine optimal edge nodes for task processing and reliable tracking. The simulation results demonstrate the superiority of the proposed UAV-based target tracking on the predefined trajectory over several existing techniques.**

***Index Terms*—Edge computing (EC), task offloading, unmanned aerial vehicle (UAV)**

I. Introduction

Unmanned aerial vehicles (UAVs) have been effectively deployed in a variety of applications during the last decade, including wireless communications for civilian, commercial, and military services [1]. Target tracking is one of the activities performed by UAVs to track movable targets in order to support a variety of military, surveillance, and mapping applications. However, UAVs have a limited power supply and limited computer capabilities. As a result, they tend to fail to execute tasks that demand extensive processing, facing significant problems in terms of computing capabilities, low latency, and inference accuracy requirements. The edge computing (EC) technique has emerged as a promising solution to address those challenges imposed on UAVs [2].

The compute-intensive operations can be offloaded to edge computing nodes by utilizing edge's computational capacity. EC has been recognised as a potential technique for reaping the benefits of heterogeneous internet of things (IoT) applications, as it can utilize diverse cloud resources such as storage and computing capabilities closer to the UAVs [3]. EC is a novel concept that places cloud servers near the mobile nodes [4].

A brief comparison of cloud computing and edge computing indicates that cloud computing has high end-to-end latency due to the distance between the edge and remote data centres, whereas edge computing has a low end-to-end latency due to its proximity to users. In cloud computing, data acquired by sensors is uploaded to the cloud, and output is given back to the desired devices, which may consume a lot of backbone network bandwidth, and can result in a delay in response. On the other hand, edge computing enables sensor data to be stored and processed on edge devices rather than in the cloud, thus it uses less bandwidth than cloud computing, fulfilling requirements such as low latency, real-time response, and reduced network traffic.

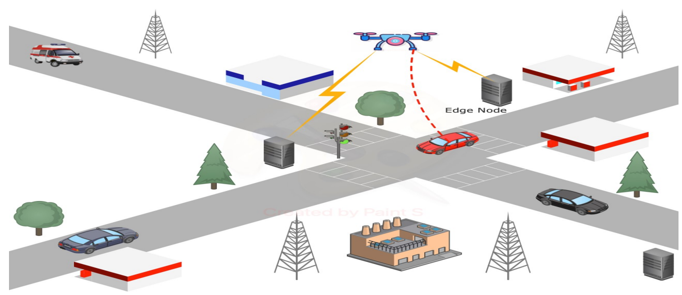

Figure 1: Network model

The quality of computing and the battery lifetime can both be enhanced by offloading computation operations to the EC server. However, the technology is infeasible in some scenarios with limited accessible infrastructure, such as disaster response, military mobility, emergency assistance, or rural locations. Therefore, unmanned aerial vehicle (UAV)-enabled EC was envisioned and developed as a viable option to address this drawback [5]. The previous studies [5], [6] have mainly focused on the computation and communication offloading of tasks, without the computation time at the UAVs. Nevertheless, the computation time cannot be neglected in a real situation. The UAV offloads computational tasks to an edge node (EN) for cooperative processing and then collects data to adjust its trajectories to follow the targets. There are two essential challenges that must be addressed in order to offer fast and efficient target tracking. The first is about dynamic tracking trajectories and adaptive edge device selection. The second issue is that a battery-powered UAV has limited resources, especially for real-time computation.

Against this background, providing a suitable solution for optimized task offloading during UAV-assisted target tracking is critical. In this article, we present a novel UAV-aided target tracking algorithm where edge nodes are selected to minimize the total cost and UAV's transmit power. We design an edge node selection algorithm by considering the capability of edge nodes for task processing and the distances between the edge nodes and the UAV. The following are the key contributions of this paper:

1) We propose an effective UAV task distribution algorithm that adjusts transmission power and selects an optimal edge node, to achieve efficient target tracking over the predefined trajectory.
2) We propose three computing strategies: local, total offloading and partial offloading.
3) The simulations are conducted to evaluate our algorithm's performance and results show that our algorithm outperforms existing works while minimizing the total cost.

The remainder of the paper is organised as follows: Section 2 presents the recent research on computation offloading in UAV-enabled EC networks. Section 3 presents our proposed model and the core modelling procedure. In Section 4, we describe the proposed algorithm. Section 5 presents a comparison of our approach against state-of-the-art solutions. Finally, concluding remarks are drawn in Section 6.

II. Related work

UAVs have attracted considerable attention from both academia and industry due to their high flexibility in deployment. UAVs have been used in a variety of wireless communication applications, including non-orthogonal multiple access networks [7], UAV-aided seamless coverage [8], mmWave communications [9], and caching [10]. MmWave is a possible approach for fulfilling the high data rate demands of 5G by providing data rates at gigabit per second. Edge-caching is an effective solution that can cope with the unprecedented growth in data traffic and reduce delivery latency by bringing content close to end-users. A previous study presented the three-dimensional (3-D) coverage performance of the cellular network-connected UAVs that function as aerial nodes [11]. UAV relaying was also reported as an important application that can enhance the coverage for communications [12].

The target tracking network for UAVs integrated with ground nodes is more efficient than the ground target tracking network. However, since the ground nodes have limited computational capability, real-time target monitoring is challenging. As a result, when the UAV in an edge environment can catch up with the target, the problem of weak processing capabilities of ground nodes may be alleviated, and the performance of the tracking system can be enhanced [13].

In comparison with cellular infrastructure-based networks, UAV-enabled EC features dependency on line-of-sight (LoS) connectivity and controlled mobility management. For example, in a previous study [6], the researchers created a dispersed deployment strategy for UAVs, which maximised the average distance between the UAV and ground node. Nevertheless, they assumed that the UAVs served all nodes with the same data rate instead of introducing variable data rates for different nodes [6]. A UAV can also be employed as a mobile cloud computing system and a UAV-mounted cloudlet [14], which provides offloading options to ground nodes. As a result, UAVs can provide fog computing even in the absence of a functioning wireless infrastructure [14]. There are limited investigations exploring the usage of UAV in target tracking process.

Some applications investigated the impact of energy consumption in UAV communication systems. The study of energy-efficient UAV communication considers throughput and UAV energy consumption [15]. However, there have been limited works in minimizing UAV energy consumption in UAV-based target tracking settings. In [16], an energy-saving technique is proposed for continuous target tracking, although this study only examined the energy consumption of the UAV during movement and ignored the energy consumption for data transmission. To the best of our knowledge, in UAV-assisted target tracking, no research has considered both the selection of edge nodes and the adjustment of transmission power.

III. Network Model and Problem Formulation

We propose a UAV-enabled EC system, as depicted in Figure 1, whereby each edge node performs task computation for the overlaying UAVs. The UAV can discover neighbouring edge nodes and save information such as their location, processing power, and other data. A UAV acquires data during UAV-assisted target tracking that must be quickly analysed and processed, and to guarantee successful tracking, the findings must be delivered back to the UAV.

The UAV can offload the computing tasks to the edge node, and subsequently the computing results can be sent back to the UAV. The UAV adjusts its position to maintain consistent tracking and to receive information of available edge nodes and offload computational tasks to the edge nodes for cooperative processing. A UAV that offloads tasks to an edge node on a regular basis for a duration of T is considered. We assume that the UAV flies at a constant altitude $H > 0$ above the ground and $\mathbf{l}_{u,m}$ is the position of UAV projected on the ground at time slot m, $\{1 \leq m \leq M\}$. We also defined T as the mission completion time, which can be divided into M time slots with a slot length of τ, $T = \tau M$. The slot length should be small enough to ensure an unchanged approximate position of the UAV during each slot. The set of UAVs defined as $\mathbf{U} \triangleq \{1, 2, ..., U\}$ are used to track the targets. We assume that there are Z edge nodes denoted as a set $\mathbf{Z} \triangleq \{1, 2, ..., Z\}$. It is assumed that a UAV can offload the tasks to an edge node $x_{z,m}$ for a duration of T and only one edge node can be selected as best edge node for serving UAV. We defined $\mathbf{X} = \{x_{z,m}, \forall z, m\}$ as the set containing the schedules of the edge nodes, where $x_{z,m}$ as a variable indicating whether edge node z is selected at time slot m. If edge node z is selected as the optimal edge node for task offloading, then $x_{z,m} = 1$ otherwise $x_{z,m} = 0$. The distance between selected edge node and UAV is defined as [17]:

$$d_{u,z}^{m} = \sqrt{H^2 + \|\mathbf{l}_{u,m} - \mathbf{l}_{z,m}\|^2} \tag{1}$$

where $\mathbf{l}_{z,m} \in \mathbf{R}^2$ is the location of edge node z selected in time slot m, $\mathbf{l}_{u,m} \in \mathbf{R}^2$ is the position of UAV u projected on the ground at time slot m, and $\|\cdot\|$ is an l_2 norm. For the communication link between the UAV and an edge node, we consider quasi-static fading model, where the channels may be changed between time slots but remain fixed at each time slot. Because the UAV flies relatively high and the probability of the UAV dispersing is minimal, thus line-of-sight (LoS) links can be established between edge nodes on the ground and the UAV [18]. The quasi-static fading model is defined as:

$$h_{u,z}^{m} = g_0 (d_{u,z}^{m})^{-2} = \frac{g_0}{H^2 + \|\mathbf{l}_{u,m} - \mathbf{l}_{z,m}\|^2} \tag{2}$$

where g_0 refers to channel power gain at 1 meter away from UAV. Assume $p_{m,z}$ is the transmission power of edge node z at time slot m. The transmit power is a variable relevant to the distance between the edge node and UAV and is used to decrease the transmission energy consumption. The transmission power can be modified to reduce transmit energy consumption, however, it is dependent on the distance between the UAV and the edge node as well as the edge node's processing capacity. The channel capacity in bits per second is stated as:

$$R_{m,z} = \frac{B}{n} \log_2 \left(1 + \frac{p_{m,z} \left| h_{u,z}^{m} \right|^2}{\sigma^2} \right) \tag{3}$$

where B shows the the channel bandwidth between UAV and edge node that can be divided into n subbands for the offloading communication, σ^2 is the variance of the white Gaussian noise (WGN) channel at the edge node, and $(\frac{p_{m,z} \left| h_{u,z}^{m} \right|^2}{\sigma^2})$ is the signal-to-noise ratio (SNR) at $d_0 = 1m$, where d_0 refers to the distance between the UAV and the edge node at 1 meter. The total transmission time and computing time of the edge node which can serve the UAV are used to calculate the task execution time as follows

$$t_{m,z}^{total} = t_{m,z}^{transmission} + t_{m,z}^{computation} \tag{4}$$

where the transmission time is calculated as:

$$t_{m,z}^{transmission} = \frac{s_m}{B \log_2 (1 + \frac{g_0 p_{m,z}}{\sigma^2 . (d_{u,z}^{m})^2})} \tag{5}$$

where s_m (bits) refers to the size of computation input data in time slot m. The computing time $t_{m,z}^{computation}$ is calculated as:

$$t_{m,z}^{computation} = \frac{s_m}{r_{m,z}} \tag{6}$$

where $r_{m,z}$ is the capacity of data processing (bytes per second) of the z-th available edge node for each task at time slot m, τ refers to the task delay tolerance, and $t_{m,z}^{total} \leq \tau$ guarantees that the UAV receives the data and adjusts time on each task as needed. For the energy component, we evaluate the energy consumed during transmission, i.e., the transmission energy $E_{m,z}$ in Joule (J) as follows:

$$E_{m,z} = t_{m,z}^{transmission} p_{m,z} \tag{7}$$

where $p_{m,z}$ is the transmit power of edge node z at time slot m. According to equation (7), the key elements that influence transmission energy usage are transmission time and transmission power. The transmit power has a significant impact on the transmission energy consumption of UAV and $E_{m,z}$ is an increasing function of $p_{m,z}$. In addition, the distance between UAV and edge node can affect transmission energy consumption. The transmission time is reduced when the UAV is offloading the tasks to a close edge node, and subsequently the energy consumption is decreased. Furthermore, the selected edge node needs to have sufficient data processing capability. Based on the data processing capabilities of edge nodes and the distance between UAV and edge nodes, we divide UAVs into three groups: $\mathbf{U}_1 \subset \mathbf{U}$, $\mathbf{U}_2 \subset \mathbf{U}$ and $\mathbf{U}_3 \subset \mathbf{U}$. The UAVs belonging to $\mathbf{U}_1$ can only carry out tasks locally since they are not near the edge node or the edge node has limited data processing capabilities. The UAVs belonging to $\mathbf{U}_2$ can only offload tasks to edge nodes, because of the limited computational resources. The UAVs belonging to $\mathbf{U}_3$ can compute the tasks locally and also they are able to offload part of the tasks to available edge node according to offloading ratio. The aim of a UAV is to minimize the total cost and it is defined as a weighted sum of the time cost and energy cost. It is measured by a total cost metric $C_{m,z}$ for executing a task.

$$C_{m,z} = \alpha E_{m,z} + \beta t_{m,z}^{total} \tag{8}$$

where α and β are weighting parameters, set in different situations accordingly (see Section V). A low cost value indicates a low-energy and execution time target tracking technique. We propose an algorithm to optimize the adjustment between transmission energy and execution time. For this purpose, the UAV's transmission power is optimized and a new adaptive scheme for edge node selection is derived based on local computing and offloading computing strategies. The problem can be formulated as

$$\min_{x_{m,z}} \sum_{m=1}^{M} \sum_{z=1}^{Z} x_{m,z} C_{m,z} \tag{9}$$

subject to the following constraints:

$$\mathcal{C}_1 : t_{m,z}^{total} \leq \tau, \forall z, m \tag{10}$$

$$\mathcal{C}_2 : \sum_{z=1}^{Z} x_{m,z} = 1, \forall z, m \tag{11}$$

where the constraint $\mathcal{C}_1$ indicates that the total execution time should be equal or less than a delay tolerance τ for task offloading. During normal target tracking, UAV needs to receive the results from edge node and make adjustment in time. The constraint $\mathcal{C}_2$ indicates $x_{m,z} = 1$, it means that at each time slot m only one edge node $z \in \mathbf{Z}$ can serve UAV for task offloading.

A) Local Computing: The computing task is executed at the UAV in the local computing scenario. Denote the UAV's CPU frequency as CPU cycles per second. The local computation delay is calculated as

$$t_l^i = \frac{c_i}{f_l^i} \tag{12}$$

where c_i is the total number of CPU cycles required to accomplish the computation for thee i-th task and s_i (bits) denotes the size of the input data related to the i-th task; f_l^i is the CPU cycle frequency of UAV for local computing (denoted as subscript "l") of the i-th task.

According to the widely adopted model [19], the energy consumed for local processing on UAV can be calculated as

$$\varepsilon_l^i = k(f_l^i)^2.c_i \tag{13}$$

where k is the energy efficiency parameter that mainly depends on the chip architecture [20], and f_l^i is the CPU clock speed. The weighted cost for local computing is defined as

$$\mathcal{O}_l^i = \theta t_l^i + (1-\theta)\varepsilon_l^i \tag{14}$$

where θ and $(1-\theta)$, $0 \leq \theta \leq 1$, indicate the UAV's preference on processing delay and energy consumption, respectively.

B) Offloading Computing: The task offloading means that the UAV can offload the computing tasks to near edge nodes. In this case, delay and energy consumption at both the edge node and via wireless link should be measured [21]. The delay for offloading (denoted as a subscript "o" below) the task to the edge node is given by

$$t_o^i = (\frac{c_i}{f_l^i} + \gamma_i(\frac{s_i}{R_i} + \frac{c_i}{f_i})) \tag{15}$$

where s_i (bits) denotes the size of computation input data for task i, γ_i is the scale coefficient i.e. $\gamma_i = s_{out}^i/s_i$, where s_{out}^i is the size of the data output from UAV, and R_i is the available data rate for the data transmitted between the UAV and edge node. The energy consumption of UAV using offloading computing is calculated as [22],

$$\varepsilon_o^i = k(f_l^i)^2 c_i + \gamma_i(P_i^m \frac{s_i}{R_i} + P_i^m \frac{c_i}{f_i}) \tag{16}$$

where P_i^m is the power consumption of UAV, when UAV sending i-th task at time slot m to edge node and staying idle while waiting for the execution results from edge node. The weighted cost for offloading computing is defined as

$$\mathcal{O}_o^i = \theta t_o^i + (1-\theta)\varepsilon_o^i \tag{17}$$

where θ and $(1-\theta)$, $0 \leq \theta \leq 1$, indicate the UAV's preference on processing delay and energy consumption, respectively.

C) Partial Computing: In the partial offloading scheme, both the UAV and edge nodes are used for computing the tasks. We define ω as the ratio of data offloaded to edge node from UAV and $1-\omega$ shows the ratio of data to be computed locally on UAV. We assume that the total data can be divided into two portions, among which ωs_i (bits) is offloaded to edge node and $(1-\omega)s_i$ (bits) is computed locally at the UAV. The total delay imposed by the partial offloading strategy (denoted with subscript "p" below) is computed as:

$$t_p^i = (\frac{c_i}{f_l^i} + \omega(\gamma_i(\frac{s_i}{R_i} + \frac{c_i}{f_i}))) \tag{18}$$

It should be mentioned that the energy consumption introduced by partial offloading is given by (ε_l^i + ε_o^i). Based on this analysis, the overall cost of UAV utilising the partial offloading strategy is calculated as:

$$\mathcal{O}_p^i = \theta t_p^i + (1-\theta)(\varepsilon_l^i + \varepsilon_o^i) \tag{19}$$

We analysed the offloading ratio ω of the partial offloading to minimize the cost combining energy consumption and execution time.

IV. Proposed Algorithm

In this section, we present the proposed algorithm for minimizing the total cost by the optimal edge node selection for task offloading based on task processing capability and the distance between UAV and edge node. The pseudo-code for the edge node selection (ENS) is summarized in Algorithm 1.

In Lines 1-9, we need to recognize local, total offloading, or partial offloading computing. In the case of offloading, the UAV needs to find the optimal edge node. In Line 16, the distances between UAV and nearby edge nodes are calculated. In Line 17, the edge nodes within 100 meters from UAV are extracted. The edge nodes with the processing capability (PC) equal to or more than 2 Mb/s are extracted in Line 18. The total cost of available edge nodes is computed by calling the procedure SELECT in Line 19. In Line 23, we set a value of ω between 0 and 1 for partial offloading. In Line 24, the best edge node is selected by calling the procedure SELECT for partial offloading. Then, ωs_i bits of data are offloaded to the best edge node for computing. Note that we set the maximum distance between UAV and edge node to 100 meters. In addition, we set the minimum capacity for the edge node to 2 Mb/s.

V. Numerical Evaluation

In this section, we present the simulation studies to evaluate the performance of the proposed algorithm. We investigate the scenario of straight flight, where the UAV flies at a constant speed from an initial location to a final destination. We consider a system with 10 edge nodes placed at various locations randomly and within an area of 500 × 500 m^2. At each time slot m, the data generated by UAV is 120 Mb. The relevant parameters for evaluating the performance of our model are summarized in Table 1. The codes and results for the experiments can be found from a GitHub link[1].

Furthermore, we consider the effects of different relative weights on task execution in terms of the cost of energy and time. We set the weights of energy cost α and time cost β equal to 0.9 and 0.1 respectively. Figure 2 shows the performance when setting different values for the two weights. We found that the cost of energy decreases when the weight on energy

[1]https://github.com/Sh-Goud/UAV-EDGE-Selection-Tracking.git

Table I: List of Simulation Parameters.

Description	Parameter	Value
System bandwidth	B	1 MHz
UAV Altitude	H	100 m
UAV transmission power	P_t	100 dBm
Simulation area	–	$500\,\mathrm{m}^2$
Radio range of the UAVs	R	500-800 m
Noise power spectral density at an edge node receiver	N_0	$-170\mathrm{dBm/Hz}$
Corresponding noise channel	σ^2	-110 dBm
Maximum transmit power of the UAV		20 dBm (0.1W)
Number of time slots	N	30
Duration of UAV flight	t_f	2 s
Total number of CPU cycles	C_i	50
Effective switching capacitance	η_c	10^{-28}
Channel power gain	g_0	-50 dB
Transmission constant	C_t	-11Db
Probabilistic SINR Threshold	SINR Threshold	-6 dB
Size of computation task input data for ith UAV (bit)	$\bar{D}$	150 Mb
Receiving threshold	Th	1.17557e-10 W
Constant speed of the UAV	V_c	10 m/s
Energy efficiency	k	10e-11

cost α increases. The same trend can be observed for β. If we increase the value of β then the time cost reduces. We also vary the weights in different situations and observe the performance changes. Figure 3 shows that, when we use a bigger α, the total cost is smaller than those in other cases. A task with a longer execution time needs more energy consumption and in turn, a bigger weight on energy consumption is needed to meet this demand. Also, in target tracking process the latency requirement is stringent and the UAV needs more time for its status adjustment. As such, we need to set bigger weight on α in target tracking process. The performance of the proposed scheme is compared with the following benchmark schemes:

1) Our scheme (Optimized): The UAV is allowed to offload the tasks to the best edge node if required. The selection of the best edge node is based on distance between edge node and UAV and processing capacity of the edge node.

2) Benchmark 1: The UAV is allowed to transmit at maximum power. The edge selection is based on the distance and the edge node nearest to the UAV is chosen as a service node.

3) Benchmark 2: In this design, the UAV selects edge nodes randomly under a fixed amount of transmit power.

Figure 4 shows a comparison between the benchmark algorithms and the proposed algorithm in terms of cost of energy and time. Since benchmark schemes do not take into consideration on the processing capability of edge nodes, these algorithms have great fluctuation on time cost. The proposed algorithm has lower energy consumption in comparison with other algorithms. Figure 5 shows that the optimized scheme archives lower total cost in comparison with other methods. The key reason behind this is that our algorithm performs the optimal edge node selection to reduce energy costs while maintaining an appropriate execution time. Figure 6 shows the average total cost in local computing, offloading computing and partial offloading. We found that the average total cost decreases in partial offloading by increasing ω. It means that with a large value of ω much data can be offloaded to the edge node. It can be seen that it is a good solution to reduce average total cost. This is due to the fact that offloading data to the edge node can reduce UAV energy consumption without incurring significant communication latency. Figure 7 and Figure 8 show the UAV's trajectories in two separate scenarios with different energy and time cost weights. Based on the proposed algorithm, the UAV selects the optimal edge node for task offloading based on task processing capability and the distance between UAV and edge node.

Algorithm 1 ENS

Input:
UAV's trajectory
Location of edge nodes
Processing capability (PC) of edge nodes
Relative weights ($\alpha \geq 0$) and ($\beta \geq 0$).
The input data size: s_i
Offloading ratio: $\omega \in [0, 1]$
Output:
Best edge node (z_{best})

if $u \in \mathbf{U_1}$ **then**
  $strategy =$ locally
end if
if $u \in \mathbf{U_2}$ **then**
  $strategy =$ total offloading
end if
if $u \in \mathbf{U_3}$ **then**
  $strategy =$ partial offloading
end if
switch $strategy$ **do**
  case locally:
    Task locally performs on UAV
    break;
  case total offloading:
    for each time slot m **do**
      Calculate the distance between edge node and UAV
      Extract $\mathbf{Z_I} = \{z_i \in \mathbf{Z}, dist(z_i, u) \leq 100m\}$
      Extract $\mathbf{Z_A} = \{z_i \in \mathbf{Z_I}, PC \geq 2MB/s\}$
      Call SELECT ($\mathbf{Z_A}$)
    end for
    break;
  case partial offloading:
    Adjust the ω
    Call SELECT ($\mathbf{Z_A}$)
    ωs_i (bits) offload to z_{best}
    break;

Select Best Edge Node

procedure SELECT($\mathbf{Z_A}$)
  for each $z_i \in \mathbf{Z_A}$ **do**
    Calculate total cost C_{m,z_i} according to (8)
    if $C_{m,z_i} < min$ **then**
      $min = C_{m,z_i}$
      $z_{best} = z_i$
    end if
  end for
  return z_{best} ▷ z_{best} is the best edge node
end procedure

VI. Conclusions and Future Directions

In this paper, an algorithm for optimal task distribution during tracking was proposed to fully utilize of the computational capability across the system. The main idea is to use an optimization technique to adjust the transmission energy consumption of UAV and accelerate task execution during normal tracking. Our simulation results demonstrate the effectiveness of the proposed algorithm for selecting an appropriate edge node during target tracking, where the UAV is used to follow

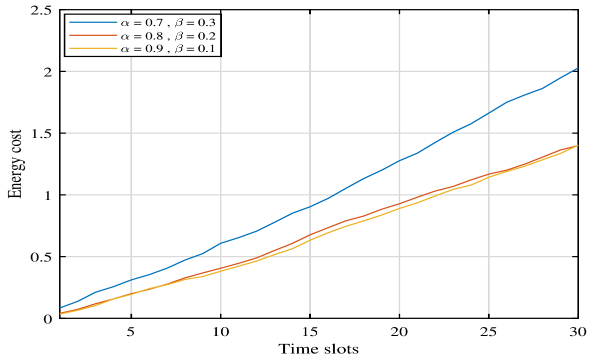

Figure 2: Energy cost vs. time slots with different α and β

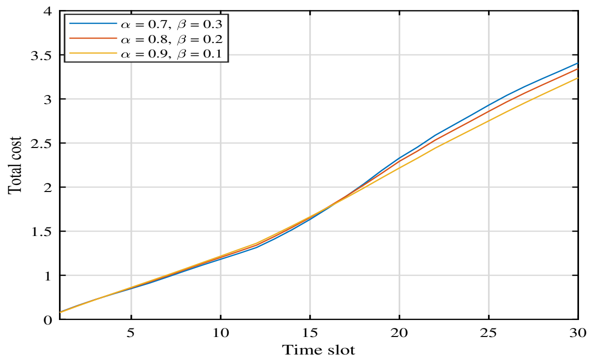

Figure 3: Total cost vs. time slots with different α and β

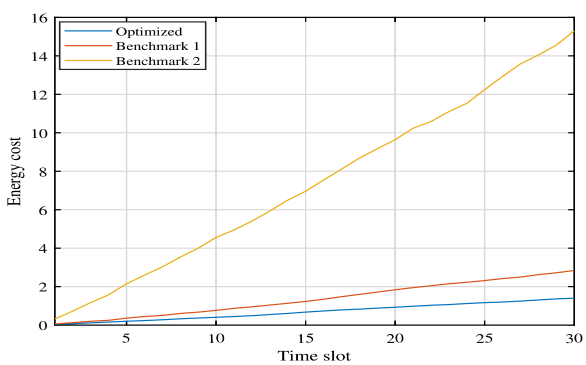

Figure 4: Comparison in terms of energy cost

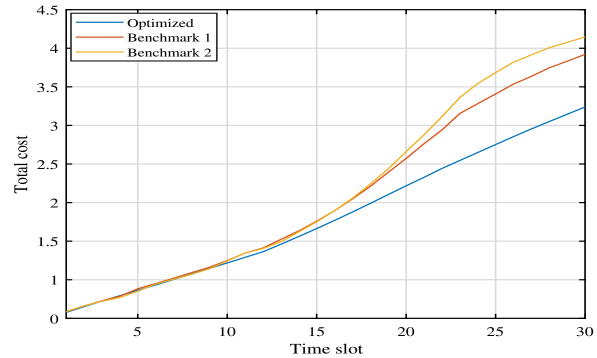

Figure 5: Comparison in terms of total cost

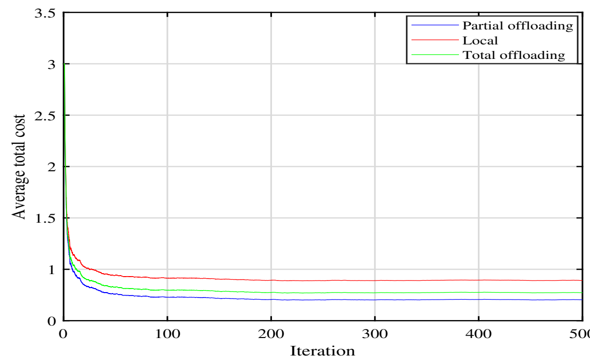

Figure 6: Average total cost

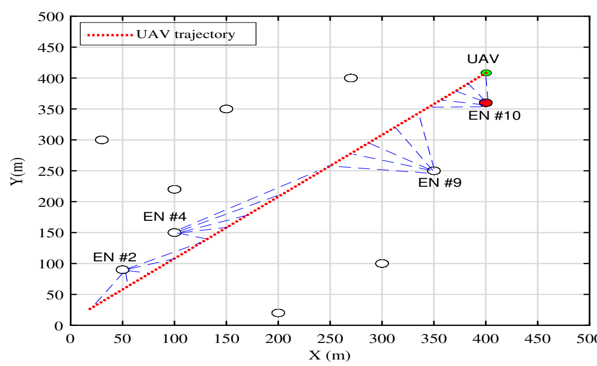

Figure 7: Trajectory 1

the given trajectory of the target. Significant enhancement in the performance was obtained using the proposed model in comparison to the baseline schemes. In the future, we will study the scenario where the target trajectory is not given to the UAV during target tracking.

ACKNOWLEDGMENT

This research is sponsored by the US Army Research Laboratory and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing under the SIGNeTS project. It is accomplished under Cooperative Agreement Number W911NF-20-2-0225. The views and conclusions contained in this document are of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, the MOD, the U.S. Government or the U.K. Government. The U.S. Government and U.K. Government are authorised to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

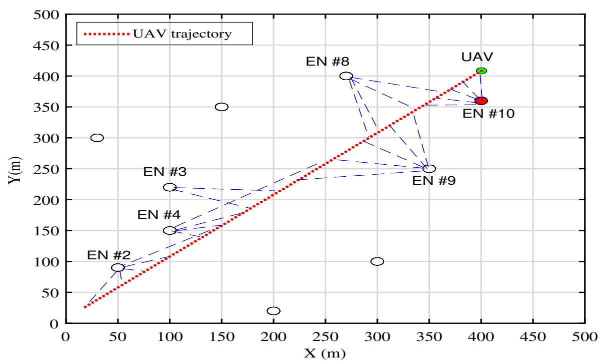

Figure 8: Trajectory 2

References

[1] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[3] S. Goudarzi, M. H. Anisi, H. Ahmadi, and L. Musavian, "Dynamic resource allocation model for distribution operations using sdn," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 976–988, 2020.

[4] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[5] J. Wang, K. Liu, and J. Pan, "Online uav-mounted edge server dispatching for mobile-to-mobile edge computing," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 1375–1386, 2019.

[6] H. Dai, H. Zhang, B. Wang, and L. Yang, "The multi-objective deployment optimization of uav-mounted cache-enabled base stations," *Physical Communication*, vol. 34, pp. 114–120, 2019.

[7] N. Zhao, X. Pang, Z. Li, Y. Chen, F. Li, Z. Ding, and M.-S. Alouini, "Joint trajectory and precoding optimization for uav-assisted noma networks," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3723–3735, 2019.

[8] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016.

[9] Y. Zhu, G. Zheng, and M. Fitch, "Secrecy rate analysis of uav-enabled mmwave networks using matérn hardcore point processes," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 7, pp. 1397–1409, 2018.

[10] M. Chen, W. Saad, and C. Yin, "Liquid state machine learning for resource and cache management in lte-u unmanned aerial vehicle (uav) networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 3, pp. 1504–1517, 2019.

[11] J. Lyu and R. Zhang, "Network-connected uav: 3-d system modeling and coverage performance analysis," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 7048–7060, 2019.

[12] S. Goudarzi, M. H. Anisi, D. Ciuonzo, S. A. Soleymani, and A. Pescape, "Employing unmanned aerial vehicles for improving handoff using cooperative game theory," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 2, pp. 776–794, 2020.

[13] L. Zhang and N. Ansari, "Latency-aware iot service provisioning in uav-aided mobile-edge computing networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10 573–10 580, 2020.

[14] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a uav-mounted cloudlet: Optimization of bit allocation and path planning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2049–2063, 2017.

[15] Y. Zeng and R. Zhang, "Energy-efficient uav communication with trajectory optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, 2017.

[16] M. Elloumi, B. Escrig, R. Dhaou, H. Idoudi, and L. A. Saidane, "Designing an energy efficient uav tracking algorithm," in *13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2017, pp. 127–132.

[17] B. Yang, X. Cao, C. Yuen, and L. Qian, "Offloading optimization in edge computing for deep-learning-enabled target tracking by internet of uavs," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9878–9893, 2020.

[18] M. Cui, G. Zhang, Q. Wu, and D. W. K. Ng, "Robust trajectory and transmit power design for secure uav communications," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 9042–9046, 2018.

[19] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 974–983, 2014.

[20] X. Gu, G. Zhang, M. Wang, W. Duan, M. Wen, and P.-H. Ho, "Uav-aided energy efficient edge computing networks: Security offloading optimization," *IEEE Internet of Things Journal*, 2021.

[21] R. Wang, Y. Cao, A. Noor, T. A. Alamoudi, and R. Nour, "Agent-enabled task offloading in uav-aided mobile edge computing," *Computer Communications*, vol. 149, pp. 324–331, 2020.

[22] M. Alsenwi, Y. K. Tun, S. R. Pandey, N. N. Ei, and C. S. Hong, "Uav-assisted multi-access edge computing system: An energy-efficient resource management framework," in *2020 International Conference on Information Networking (ICOIN)*. IEEE, 2020, pp. 214–219.