Multiple UAVs Trajectory Optimization in Multi-Cell Networks with Adjustable Overlapping Coverage

Jongyul Lee, Student Member, IEEE and Vasilis Friderikos, Member, IEEE

Abstract—Trajectory optimization of Unmanned Aerial Vehicles (UAVs) operating as flying base stations (FBSs) evolved as a novel integration component in beyond 5G (B5G) networks and has recently received significant research attention. Notably, the vast majority of previous research has mainly concentrated on the case of a single terrestrial macro base station (BS) which is used as a depot for multiple FBSs. In this paper, we focus on the more general use case where multiple FBSs located at different macro-BSs used as a depot serve ground users (GUs) at cluster points (CPs). To this end, we formulate the FBSs trajectory optimization problem using a mixed integer linear programming (MILP) formulation with the aim to minimize total travel time (TTT) of the FBSs in a multi-cell network in which their cell coverage or boundary is adjustable for the FBSs deployment; creating in that sense virtual cells for the FBSs. Furthermore, heuristic algorithms are proposed to provide competitive solutions and reduce the computational time in view of the curse of dimensionality of the original problem. Numerical investigations reveal that the proposed FBSs path planning optimization solutions decrease the TTT and increase the efficiency of offloading collected data for the FBSs deployment with gains up to approximately 23% and 19% respectively compared to nominal schemes that consider the pre-defined coverage range of the cells or no cell boundaries. Aside from the above, compared to previously proposed nominal strategies, the proposed schemes achieve an almost 27% improvement in terms of fairness (Jain’s index) on the FBS traveling time.

Index Terms—Wireless Networks, B5G, 6G, Mixed Integer Linear Programming (MILP), Unmanned Aerial Vehicle (UAV), Multi-cell, Trajectory optimization.

I. INTRODUCTION

U

MANNED aerial vehicles (UAVs) acting as flying base stations (FBSs) have recently gained increasing research attention and commercial popularity due to their inherent flexible and on-demand ease of deployment to enhance the performance of terrestrial mobile network infrastructures in a multifaceted manner. The FBSs can be considered as a natural extension to heterogeneous network (HetNet) which is represented by small cells, e.g. a pico- and femtocell, in addition to the macro-base stations (BSs). The HetNet can assist the existing network to mitigate the increasing data traffic demand and improve network efficiency and flexibility by providing on-demand high capacity connectivity and reliable service in the macro-cell network.

The increased requirements for data consumption from end-users together with an increased number of connected ‘things’ in beyond 5G (B5G) networks pose significant challenges in network provisioning. This is especially true in crowded urban areas with high spatio-temporal traffic variations. On the other side of the spectrum, rural and isolated environments have inherently low capacity installed and thus it is challenging to provide an adequate service in cases of sporadic increase of the traffic demand. Therefore, the concept of UAV-assisted B5G network in the form of an FBS has been spotlighted with the potential to augment the terrestrial B5G network operations in the above cases and for a plethora of different use cases such as information dissemination, on-demand hot spot data coverage and data collection for Internet of Things (IoTs) to name just a few. It is technically feasible for UAVs to be deployed for a significant amount of flying time. In that respect, considering the use case of multiple FBSs that are located and dispatched from terrestrial BSs (acting as their depot) is of significant importance in order to provide various advanced services in an agile on-demand manner whilst improving both the network capacity and the coverage of the terrestrial macro-BSs. More specifically, they can allow an increased lifetime of the IoT sensors on the ground since energy consumption is reduced due to the close proximity between the FBS, that is used to offload data, and the IoT sensors on the ground.

Undoubtedly, the path planning of multiple FBSs is one of the most crucial tasks in the macro cellular UAV-assisted B5G networks, as will heavily define the underlying network performance. Based on the trajectory (path) planning, FBSs serve a number of different ground users (GUs) and IoT sensors in a defined order and thus result in an efficient aggregated travelling time and overall energy efficiency. Worth pointing out that the UAVs acting as FBSs consume a significant amount of energy for both flying and hovering, and their mission time (i.e., flying time) is considerably restricted by the limited available on-board energy. In this regard, a trajectory optimization problem that amalgamates the well-known Traveling Salesman Problem (TSP) and Vehicle Routing Problem (VRP) is required for the efficient FBS path planning.

This research work was presented in part at the IEEE Wireless Communications and Networking Conference (WCNC), May 25-28, 2020, DOI: 10.1109/WCNC45663.2020.9120711.

J. Lee is with Ericsson, System R&D, Seoul, Republic of Korea, and V. Friderikos is with the Department of Engineering, Centre of Telecommunications Research, King’s College London, London, WC2R 2LS, U.K. (e-mail: jongyul.lee@ericsson.com, vasilis.friderikos@kcl.ac.uk).

1. Hereafter, the terms of ‘UAV’ and ‘FBS’ are used interchangeably.
A. Related Works

Due to the plethora of potential applications of UAVs being used as FBSs, as eluded in the previous section, the performance of the underlying trajectories of UAVs has recently attracted a great deal of interest [11]–[22]. To this end, various TSP and VRP type of models for the UAV trajectory optimization problem have been previously proposed [13]–[15], [23], [24].

The work in [11], [12] formulate a UAV trajectory optimization framework aiming to minimize UAV’s mission completion time, whilst satisfying a defined volume of data uploading from sensor nodes and maintaining a target quality of connectivity in the network. In a similar setting, the authors in [13]–[15] proposed a trajectory design of UAVs inspired by the TSP with the main focus of consideration on energy efficiency. Also, the authors in [16] investigate power control of UAV acting as a mobile gateway for the data collection service to minimize the energy consumption of a UAV while meeting the QoS requirement. Furthermore, in [17]–[20], the UAV trajectory optimization problem in cellular network is detailed with the emphasis on maximizing the throughput of the end-users. In [17], the authors studied a mobile relaying technique with a cache-enabled UAV at the edge cell in a multi-cell network to maximize the aggregate rate of all cell edge users. In [18], the authors present a UAV trajectory optimization for data offloading to maximize throughput of users at the edge cell area in a multi-cell network whilst minimizing the co-channel interference between BSs and FBSs. Differentiated from [17], [18], the work [20] investigated a trajectory optimization problem with a multiple UAVs setting departed from multiple charging stations to maximize the data rate of the cell-edge mobile users. In [19], the authors investigated a multi-drone cell trajectory planning and a resource allocation problem, considering multiple BSs communication coverage with focus on serving high mobility users while maximizing throughput over a geographically large network area. Moreover, the work in [21] formulated a trajectory optimization problem to maximize the number of IoT devices that are covered in the network while guaranteeing the minimum amount of data uploaded from each served device within a pre-defined given deadline. In [22], the authors provided particle swarm optimization (PSO) and genetic algorithm (GA)-based UAV trajectory algorithms for the dynamic deployment of a single FBS for mobile users, which outperforms the exhaustive search space algorithm and the static UAV deployment method in terms of both the underlying execution time.

Meanwhile, the work in [23] investigated how to design a UAV trajectory based on the TSP with the aim to minimize the task completion time, whilst considering the UAV’s battery capacity. In [24], the authors worked on a trajectory optimization problem of a UAV by exploiting the VRP type of formulations for delivery and emergency response scenarios, whilst considering underlying hardware capabilities such as the available battery and payload weight. In similar use case scenarios to the previous mentioned research, the works in [12], [20], [23], [24] explicitly considered recharging of FBSs at a macro-BS used as a depot and a recharging station after completing serving GUs and IoT sensors.

The majority of the aforementioned research contributions focused on the UAV trajectory optimization in a single-cell scenario [11]–[16], [19], [21]–[24], where the macro-BS acts as a depot for the UAVs. A limited, but closely related set of prior works such as in [17]–[20] addressed the UAV-aided networks considering a multi-cell (or multiple initial points) wireless network. However, the works [16]–[18] focused on the case of a single UAV operating only at the edge of adjacent cells. Also, the closely related works in [16]–[20] are not considering the underlying TSP (or VRP)-like UAV trajectory optimization problem, i.e. a Hamiltonian path, under a multi-cell setting. Yet, for a realistic scenario, it is inevitable to consider a depot (which can be a macro-BS) or a recharging station as the FBS requires recharge/refuel for providing iterative service (i.e. iterative mission of flying in FBS deployment) based on the spatio-temporal demand from the GUs in the network. Besides, it is crucial to take into consideration the FBS deployment in a multiple macro-BSs setting where FBSs can be hosted in different macro-BSs (depots) [4], [5]. Therefore, we explicitly investigate the multiple FBSs trajectory planning optimization problem in a multi-cell wireless network (i.e. multiple depots). In that case the coverage area of the FBS might be different from the coverage area of the macro-BS that the FBS uses as a depot; hence, we have the creation of virtual adjustable coverage areas for FBSs. Moreover, we design a Hamiltonian path-based UAV-enabled trajectory planning in conjunction with the adjustable coverage area per FBS as described above.

To visualize the above, we illustrate a schematic diagram of the proposed UAV-assisted multi-cell B5G network in Fig. 1.

We define a terrestrial plane that represents the deployment of the macro-BS and the GUs (cluster points) as well as an aerial plane that represents the FBS deployment. In Fig. 1(a), the deployment of FBSs can only operate within the restricted cell boundary where the FBSs use the macro-BS (depot). In Fig. 1(b), the coverage area, which defines the region of operation of the FBSs, of each cell is virtually extended. Hence, an overlapping region of operation for the FBSs is generated between the cells. In this case, the whole FBSs trajectories may be altered due to the different virtual cell coverage areas. This motivates us to propose a novel FBSs trajectory optimization in a multi-cell network with the virtual cell coverage for the
FBS deployment.

B. Contributions

The novelty and the main contributions of the work presented hereafter can be summarized as follows:

- We formulate a multiple FBSs hosted on multiple macro-BSs trajectory planning optimization problem operated as a Hamiltonian path using a mixed integer linear programming (MILP) by considering a nominal multi-cell mobile network model.

- The proposed formulation provides a novel virtual cell with adjustable coverage (boundary) for the FBSs deployment, where the virtual cell coverage forms separated nodes to be served. Furthermore, the proposed framework explicitly models time windows for FBSs serving different clusters that allow a different level of quality of service (QoS) to be implemented. The goal of the proposed framework is to minimize FBS’s traveling time in the multi-cell network.

- We also present three flavours of competitive heuristic algorithms employing Local Search Technique (LST) to reduce the computational complexity and implement the virtual cell with the adjustable coverage in a multi-cell setting since the proposed optimization problem is a generalization of a well-known NP-hard problem [10].

- Numerical investigations confirm that the proposed optimal schemes and heuristic algorithms outperform baseline solutions in terms of fairness, the efficiency of offloading collected data, total traveling time (TTT) and mission completion time of the FBSs.

The rest of the paper is organized as follows. In Section II, we introduce the underlying system model of the multi-cell UAV aided mobile network and in Section III, we present two types of MILP formulations. Then, in Section IV, we outline a set of competitive heuristic algorithms, which offers real-time decision making due to low computational complexity. Following that, in Section V, we provide numerical investigations to verify the effectiveness of the proposed schemes. Finally, conclusions are drawn in Section VI.

### TABLE I

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>Set of vertices</td>
</tr>
<tr>
<td>V_c</td>
<td>Set of CPs</td>
</tr>
<tr>
<td>V_d</td>
<td>Set of macro-BSs</td>
</tr>
<tr>
<td>C</td>
<td>Set of cells</td>
</tr>
<tr>
<td>Ψ</td>
<td>Set of cells</td>
</tr>
<tr>
<td>ζ_B</td>
<td>Set of CPs in the overlapping areas of two cells</td>
</tr>
<tr>
<td>ζ_Q</td>
<td>Set of CPs in the overlapping areas of three cells</td>
</tr>
<tr>
<td>Ξ_f,i</td>
<td>Achievable throughput rate at CP i for GU f</td>
</tr>
<tr>
<td>s_i</td>
<td>Limited number of deployed FBSs at macro-BS i</td>
</tr>
<tr>
<td>w_i</td>
<td>Arrival time of an FBS at CP i</td>
</tr>
<tr>
<td>r_i</td>
<td>Start time for QoS at vertex i for an FBS</td>
</tr>
<tr>
<td>l_i</td>
<td>Closing time for QoS at vertex i for an FBS</td>
</tr>
<tr>
<td>v</td>
<td>Velocity of FBS</td>
</tr>
<tr>
<td>d_{ij}</td>
<td>Distance between vertex i to vertex j</td>
</tr>
<tr>
<td>t_{ij}</td>
<td>Traveling time of an FBS from vertex i to vertex j</td>
</tr>
<tr>
<td>λ</td>
<td>Sufficiently large integer number</td>
</tr>
<tr>
<td>s_{f,i}</td>
<td>Service time that an FBS consumes for GU f at CP i</td>
</tr>
<tr>
<td>q_{f,c}</td>
<td>Length of the requested data in bits</td>
</tr>
<tr>
<td>H</td>
<td>Side length of a given hexagon cell</td>
</tr>
<tr>
<td>x_{ijk}</td>
<td>Decision variable for given vertex i, j and FBS k</td>
</tr>
<tr>
<td>y_{ijc}</td>
<td>Auxiliary index that creates an feasible or infeasible FBS route for given vertex i, j and cell c</td>
</tr>
<tr>
<td>i</td>
<td>Index of vertices</td>
</tr>
<tr>
<td>k</td>
<td>Index of an FBS</td>
</tr>
<tr>
<td>e</td>
<td>Index of a cell</td>
</tr>
<tr>
<td>f</td>
<td>Index of a GU</td>
</tr>
<tr>
<td>ψ_c</td>
<td>Index of cell c</td>
</tr>
</tbody>
</table>

**Notation:** The notation used throughout the paper is reported in Table I.

### II. System Model

A. Scenario

The overarching use case scenario relates to the UAV-assisted B5G network as depicted in Fig. 1 which includes the procedure of a nominal envisioned operation of multiple FBSs hosted at terrestrial macro-BSs. This procedure is briefly described below:

- Without loss of generality, a typical hexagon cell layout is assumed and three hexagon cells are represented as

---

Fig. 1. An overview of UAV assisted B5G network in a multi-cell network. In Fig. 1(a), the FBSs trajectories are limited within respective cells due to the solid coverage. Whereas, In Fig. 1(b), the CPs at the edge of the cells are overlapped due to the adjustable (extendable) cell coverage and therefore the FBSs trajectories are changed.

---
a typical tier in a multi-cell network. Via the fronthaul link, the cells are connected to edge clouds (or the cloud controllers) where information about cluster points (CPs), e.g. their locations, can be collected and processed.

- Each B5G terrestrial macro-BS is located at the center of each cell. A set of FBSs is allocated at the macro-BSs which act as depots for those FBSs.
- A set of CPs are randomly distributed within a pre-defined geographical area in the multi-cell wireless network. A CP can be defined as the center of a cluster of nodes (GUs) including IoT sensors, where an FBS hovers to serve the GUs until the FBS completes its service. It is assumed that there are a number of GUs needed for the UAV-assisted service, scattered in the vicinity of the CP at a busy time of the network. All macro-BSs share the location of all CPs through the cloud controllers.
- The FBSs are dispatched from each macro-BS and serve at CPs. Then, they return to the original macro-BS to replenish their battery/fuel and/or to offload collected data. Each FBS begins to serve the GUs by hovering once arriving at the serving CP.
- The FBS serves the GUs as an uplink transmission for a pre-defined time period, which is also called a service time. Once the FBS completes the service, it moves to the next CP. When the FBSs visited all assigned CPs, then the FBSs return to the respective macro-BS (depot).
- A single FBS visits each CP only once in order to avoid duplicating the service, the possibility of collisions, and interference among FBSs.
- We assume that all FBSs fly with the same constant velocity denoted as \( v \) m/s.

### B. Network Model

The UAV-assisted B5G network is modeled as an undirected graph \( G = (\mathcal{V}, \mathcal{A}) \), where \( \mathcal{V} = \mathcal{V}_c \cup \mathcal{V}_d \), \( \mathcal{V}_c = \{1, 2, \cdots, N\} \) denotes a set of CPs and \( \mathcal{V}_d = \{1, 2, \cdots, M\} \) denotes a set of macro-BSs in a 2-dimensional Cartesian coordinate system. \( \mathcal{A} = \{(i, j) : i, j \in \mathcal{V}, i \neq j\} \) is a set of links (routes) with the weight denoted as \( d_{ij} \), i.e. the Euclidean distance. The travel time that FBSs travel between vertices \( i \) and \( j \), denoted as \( t_{ij} \) and it can be calculated by using the distance of the link with the FBS velocity \( v \), i.e. \( t_{ij} = d_{ij} / v \). A set of FBSs is denoted by \( \mathcal{K} = \{1, 2, \cdots, K\} \). Let \( \mathcal{X} = \{\psi_1, \psi_2, \cdots, \psi_M\} \) indicate a set of cells. Each cell includes a macro-BS and a set of \( M \) randomly distributed CPs. Let \( \mathcal{F} = \{1, 2, \cdots, NF\} \) denote the set of GUs (IoT sensors). Also, let \( s_{f,i} \) denote the required service time where a GU \( f \) at CP \( i \) requires an FBS to serve.

### C. Transmission Model

We assume an uplink scenario. The FBS flies at a constant altitude \( H \) in meters and \( q_k \in \mathbb{R}^{2 \times 1} \) denotes the FBS trajectory projected onto the horizontal plane, i.e. the location of CP \( i \). The horizontal location of the GU \( f \in \mathcal{F} \) at CP \( i \) is denoted as \( w_{f,i} \in \mathbb{R}^{2 \times 1} \). Let \( P_t \) denote the transmit power of a GU when the FBS schedules the service for the GU.

4The word 'vertex' is used to denote both the 'macro-BS' and the 'CP'.

The achievable throughput rate \( \Xi_{f,i} \) in bits per second (bps) between the UAV and the GU \( f \) at CP \( i \) is represented as,

\[
\Xi_{f,i} = B \log_2 \left( 1 + \frac{\gamma_0}{(H^2 + ||q_k - w_{f,i}||^2)^\alpha} \right),
\]

where \( B \) is the communication bandwidth in hertz (Hz), \( \gamma_0 \triangleq P_t \beta_0 / (\sigma_0 \Gamma) \) is the reference SNR where \( \beta_0 \) is the path loss at the reference distance, \( \sigma_0 \) is the noise power at the receiver, and \( \Gamma \) is a constant accounting for the gap from the channel capacity. Also, \( \alpha \) denotes a path loss exponent referred in [25]. The parameters used in Eq. (1) are referred for further details in Table II. Let \( Q_{f,i} \) denote the length of the requested data in bits for GU \( f \) at CP \( i \) and therefore the service (i.e. the required transmission) time denoted as \( s_{f,i} \) is derived by \( s_{f,i} = Q_{f,i} / \Xi_{f,i} \).

### III. Mathematical Programming Formulation

#### A. UAV Trajectory Planning Formulation with Time Windows in a Multi-Cell Network (No Cell Boundary)

In this section, based on the discussed scenario and aforementioned system model, we detail a proposed MILP formulation for path optimization with multiple FBSs, aiming at minimizing the total FBSs flying time, whilst taking into account a bounded cell setting. To this end, the binary and continuous decision variables of the proposed model can be defined as follows,

\[
x_{ijk} = \begin{cases} 
1, & \text{if an FBS } k \text{ uses a link between vertex } i \text{ and vertex } j \\
0, & \text{otherwise}
\end{cases} \quad (2)
\]

\[
w_i \geq 0, \quad \text{arrival time of an FBS at vertex } i \quad (3)
\]

We then enumerate several constraints to create an FBSs trajectory and time windows in the multi-cell network as follows,

\[
\sum_{i \in \mathcal{V}} \sum_{k \in \mathcal{K}} x_{ijk} = 1, \quad \forall j \in \mathcal{V}, \quad (4a)
\]

\[
\sum_{j \in \mathcal{V}_c} \sum_{k \in \mathcal{K}} x_{ijk} \leq \delta_i, \quad \forall i \in \mathcal{V}_d, \quad (4b)
\]

\[
\sum_{j \in \mathcal{V}} x_{ijk} = x_{ijk}, \quad \forall k \in \mathcal{K}, \quad \forall i \in \mathcal{V}. \quad (4c)
\]

The constraints in (4a)–(4c) relate to the FBS trajectory planning with multiple FBSs in a multi-cell setting. The constraint in (4a) guarantees that the FBS \( k \) travels from the vertex \( i \) to the vertex \( j \). The constraint in (4b) ensures that the number of FBSs defined as \( \delta_i \), which will depart and return to the macro-BS \( i \), is limited. The constraint in (4c) guarantees that the FBS \( k \) leaves and arrives at a determined vertex, which also means flow conservation of the FBS route.

\[
e_0 + t_{ij} \leq w_j + \lambda (1 - x_{ijk}), \quad \forall k \in \mathcal{K}, \forall i \in \mathcal{V}_d, \forall j \in \mathcal{V}_c, \quad (5a)
\]

\[
w_i + s_{f,i} + t_{ij} \leq w_j + \lambda (1 - \sum_{k \in \mathcal{K}} x_{ijk}), \quad \forall i, j (i \neq j) \in \mathcal{V}_c, \forall f \in \mathcal{F}, \quad (5b)
\]
where $K$ is a sufficiently large integer number utilized as the big-M activated (linked) as illustrated in Fig. 2. Note that the constant $\lambda$ is a sufficiently large integer number utilized as the big-M method [26]. The constraint in (5a) guarantees the earliest time when the FBS $k$ starts departing from the macro-BS $i$; $c_0$ is the mission start time for FBS deployment at a macro-BS. The $t_{ij}$ indicates the traveling time when the FBS $k$ travels from the vertex $i$ to $j$, and the $w_j$ is the arrival time of the FBS $k$ at the vertex $j$. The constraint in (5b) ensures that the FBS $k$ is arranged to arrive at the CP $i$ and also serve for the service time $s_{f,i}$ for the GU $f$, and then travel from the CP $i$ and arrive at next CP $j$. The constraint in (5c) guarantees that the FBS $k$ returns to the macro-BS before the maximum window time $l_0$. Note that the constraints in (5a)–(5c) guarantee no sub-tours [27] by Miller-Tucker-Zemlin (MTZ) formulation [28]. The constraints in (5d)–(5e) allow to obtain the exact arrival time of FBSs at each CP as well as allow to calculate the TTT. The constraint in (5f) guarantees that the FBS serves GUs at a CP $i$ within the closed time window denoted as $[c_i, l_i]$, i.e. QoS support.

Based on the above constraints, the mathematical program for the multiple FBSs path optimization with time windows, which also takes into account multiple macro-BSs, can be formulated as follows,

\[
\begin{align*}
(P1), \text{OUT-MNB:} \quad & \min \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} t_{ij} w_{ij} \\
& \text{s. t. } \quad (4a) - (5c), \quad (5a) - (5f), \\
& \quad x_{ijk} \in \{0, 1\}, \quad \forall k \in K, \quad \forall i, j (i \neq j) \in V, \\
& \quad w_i \geq 0, \quad \forall i \in V,
\end{align*}
\]

(6)\n
\[
\text{Trajectory in Multi-cell with No Boundary (OUT-MNB), is to minimize TTT for all FBSs trajectory. The constraints in (7) and (8) indicate that the decision variable } x_{ijk} \text{ is binary and the decision variable } w_i \text{ is greater or equal than zero. We note that in the problem (P1), there are no serving cell boundaries for the FBSs. In other words, there are no imposed restriction where different FBSs will be operating whilst they are hosted at a given macro-BS.}
\]

B. Modelling Multi-cell with Solid and Adjustable Coverage

In this section, we design a multi-cell setting with an adjustable coverage acting as a boundary whilst finding a global optimal FBS trajectory. Fig. 3 shows the two typical cells with respective virtual coverage according to the range of operation of FBSs. From the macro-BS 6 and 7, the respective cell coverage is bordered where each coverage is called solid coverage (SC) and can be extended where each coverage is called extendable coverage (EC). As can be seen in the figure, the vertices 1 to 7 are on the dotted straight line segment. Both two macro-BSs are located in the mid center of each cell $\psi_A$ and $\psi_B$. Also, the CP 2 and 4 are located on the EC boundary and hence both cell coverages are overlapped between them. Whereas, the CP 3 is located right on the SC boundary where the two coverages are bordered. In other words, the SC is not extended. Also, the CP 1 and 5 are located outside of each macro-BS. $R$ indicates a distance between the macro-BS 6 and the CP 2, and the macro-BS 7 and the CP 4. $\tau$ represents a relatively very short distance between CPs, i.e. between the vertices 1 and 2, 2 and 3, 3 and 4, and 5 and 7. Based on these preliminaries, we prove that the FBSs trajectory optimization in a multi-cell network under the SC model does not provide a global optimal solution via the following Lemma[1] with Fig. 4.

**Lemma 1.** A multi-cell network with FBSs under the SC model does not provide a global optimal solution to the trajectory optimization problem.

\[\psi_A \]

\[\psi_B \]

\[\text{In order to help the readability for the concept of SC and EC model, it is worth referring to Fig. 4.}\]
Proof. Without loss of the generality, we assume that the FBS velocity and the service time when the FBS serves GUs at CPs are identical. Also, two different FBSs are employed to be dispatched from macro-BSs 6 in $\psi_A$ and 7 in $\psi_B$ respectively. In the SC model, since the CP 3 is located right on the SC boundary, a single FBS departing from the macro-BS 6 in $\psi_A$ or 7 in $\psi_B$ will only visit the CP 3 and hence another FBS cannot visit it according to the scenario. Whereas, in the EC model, a single FBS visits the CP 2, 3 and 4 at once, which are incidence vertices, i.e. overlapped CPs due to the duplicated coverages. In other words, since the CP 2, 3 and 4 are overlapped in both coverages of the cell $\psi_A$ and $\psi_B$, if the overlapped coverage only belongs to either $\psi_A$ or $\psi_B$, all FBSs trajectory may be changed and they in turn lead to a different optimal gain, i.e. multiple Hamiltonian paths with incidence vertices. For instance, in the SC model, one FBS would travel in the order of 6-1-2-3-6 as the shortest path and another FBS would travel in the order of 7-5-4-7. In this case, gains of the FBSs trajectory with the SC and EC models can be calculated by following as,

$$G_{SC} : \tau + (\tau + R) + \tau + (\tau + R) + R = 4R + 6\tau, \quad (9)$$

$$G_{EC} : \tau + (\tau + R) + \tau + \tau + (2\tau + R) + \tau + \tau = 2R + 8\tau, \quad (10)$$

where $G_{SC}$ and $G_{EC}$ represent the gain of the FBSs trajectory with the SC and the EC model respectively. The gains of FBSs trajectory can be subtracted from one another as given by,

$$G_{SC} - G_{EC} = 2R - 2\tau. \quad (11)$$

Since $R$ and $\tau$ are positive numbers, Eq. (11) can be reformulated as,

$$0 \leq 2R - 2\tau, \quad \tau \leq R. \quad (12)$$

Hence, $G_{SC}$ does not provide a global optimal solution if $\tau$ is less or equal to $R$. $\square$

To establish the formulation providing a global optimal solution, we note that special attention should be placed in the FBSs trajectory for the case of the EC model. As the toy example is shown in Fig. 4, there are threefold levels of the cell coverage: 1) SC model, 2) EC model in which the Hamiltonian path is not allowed by the scenario, 3) EC model with a global optimal solution. Fig. 4(a) shows the cell coverage $\psi_A$ and $\psi_B$ with SC model, whilst Fig. 4(b) and Fig. 4(c) depict the cell coverage with EC model. In Fig. 4(a), since the FBSs trajectory is constrained by the boundaries of the given cells, i.e. due to the SC model, the gain may be deemed as a local optimal solution. In contrast, in Fig. 4(b), even though the EC model is constructed, the solution is not valid since the FBSs trajectory is duplicated through the CP 2 and 3, where the Hamiltonian path is not allowed. However, in Fig. 4(c), none of the FBSs trajectory is overlapping and hence this type is our ultimate proposed EC model and a global optimal solution can be obtained.

Hence, to construct a multi-cell, i.e. with both SC model and eventually EC model, which allows for the Hamiltonian path and attaining a global optimal solution, we first introduce a set of an auxiliary matrix values denoted as $Y = \{y_{ijc} \mid \forall i, j \in \mathcal{V}, \forall c \in \Psi\}$ as given by,

$$Y = \begin{bmatrix} y_{i11} & y_{i12} & \cdots & c \\ \vdots & \ddots & \cdots & \cdots \\ y_{i11} & y_{ijc} & \cdots & y_{ij1} \end{bmatrix}$$

where $c = \psi_c$ represents a cell. The matrix $Y$ is predefined by the location of CPs including the macro-BS and acts as clustering the CPs in the first tier; the terrestrial plane, whilst the variable $x$ plays a role in finding the best route for FBSs in the second tier; the aerial plane. If a link (route) between vertex $i$ and vertex $j$ is within the cell $c$, then $y_{ijc} = 1$, otherwise $y_{ijc} = 0$, i.e. $y$ is a binary variable. Fig. 5 indicates virtual layers of the multi-cell network with the decision variable $x$ and the auxiliary index $y$. In Fig. 5, the blue and red lines denote the potential feasible link, i.e. the case of $y_{ijc} = 1$, in the $\psi_A$ and $\psi_B$ respectively. On the other hand, the green dotted lines denote the potential infeasible link, i.e. the case of $y_{ijc} = 0$, since they deviate from the coverage of each cell. In summary, the auxiliary index $y$ allows the range of FBSs deployment in a given cell to be confined by capturing the feasible links in a different dimension.

C. UAV Trajectory Planning Formulation with Time Windows in Multi-Cell Network of Adjustable Coverage

By using the auxiliary matrix $Y$, we subsequently provide a MILP formulation of the FBSs path planning optimization problem with time windows in a multi-cell network allowing for the SC and EC model. The derived optimization problem called optimal UAV trajectory in a multi-cell with solid coverage (OUT-MSC) and extendable coverage (OUT-MEC) is formulated as follows,

(P2), OUT-MSC/MEC: \[
\min_{W, X} \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} \sum_{(i \neq j)} t_{ij} x_{ijk} \]

(P2), OUT-MSC/MEC: \[
\min_{W, X} \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} \sum_{(i \neq j)} t_{ij} x_{ijk} \]
Fig. 5. The FBSs trajectory in virtual layers of the multi-cell network illustrated by the decision variable \( x \) and the auxiliary index \( y \).

Fig. 6. Illustration of graph coloring and CPs of the typical multi-cell model for FBSs deployment according to the side length \( L_H \), (a) SC model, (b) EC model. Note that the purple CP in \( \zeta_Q \) is in 2-colourable graph area, whilst the yellow CPs in \( \zeta_B \) are in 2-colourable graph area.

\[
x_{ijc} \leq y_{ic}, \quad \forall k \in K, \quad \forall i, j (i \neq j) \in \mathcal{V}, \quad \forall c \in \Psi. \quad (15)
\]

The constraint in (15) imposes that the variable \( x \) should be less or equal than the auxiliary matrix value \( y \), which enables FBSs deployment confined in a certain boundary (coverage), i.e. SC and EC models.

Therefore, the problem (P2) corresponds to minimizing the TTT of FBSs trajectory in given cells, which is feasible to gain a global optimal solution.

IV. PROPOSED SET OF HEURISTIC ALGORITHMS

Finding optimal solutions suffers from the curse of dimensionality problem since the complexity of the underlying mathematical program resembles the well-known TSP and VRP, which are both known to be NP-hard \(^6\). This motivates us to propose alternatives to optimal solutions. In this section, we present a set of local search heuristic algorithms that is specifically designed for the multi-cell network with the EC model to find competitive solutions to the FBSs trajectory planning problem. The goal of the algorithms is to improve the total cost in order to find competitive FBS paths in the multi-cell network where the cell coverage within which the FBS can travel is overlapped.

\( R(k) \):

1: \( R(k) \): \( 8 \rightarrow 2 \rightarrow 1 \rightarrow 4 \rightarrow 8 \)
2: \( R(2) \): \( 8 \rightarrow 2 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow 8 \)
3: \( R(3) \): \( 10 \rightarrow 3 \rightarrow 7 \rightarrow 10 \)

(a) 2-Opt.
(b) Relocate.
(c) Swap.

Fig. 7. A toy example illustrated with a set of string indices for the FBSs trajectory (i.e. a multi-cell with the EC model) according to the number of FBSs, (a) Initiative string indices \( R(k) \) and the corresponding FBSs trajectory, (b) A newly subsequent \( R(k) \) and FBSs trajectory after executing the Local Search Technique (LST).

Algorithm 1 Randomized distribution with randomized local search (RD-RLS)

1: \( I_{RD}, \Theta \) \( \leftarrow \) Initialize iteration number, parameters in Table II
2: **First tier:**
3: 3: Generate \( \Psi, K, \mathcal{V} \)
4: 4: \( \hat{R}(K) \) \( \leftarrow \) Generate string indices in \( K \) rows
5: 5: \( \hat{R}(K, \mathcal{V}, \Psi, \Theta) \) \( \leftarrow \) Insert CPs and macro-BSs that belong to each \( \psi \) into the string indices in a random distributed manner
6: **Second tier:**
7: 7: \( R(K, \mathcal{V}, \Psi, \Theta) \) \( \leftarrow \) Generate UAVs trajectories with \( (K, \mathcal{V}, \Psi, \Theta) \)
8: 8: \( \Delta \) \( \leftarrow \) Compute time cost for \( \{ \hat{R}(K, \mathcal{V}, \Psi, \Theta) \} \)
9: 9: \( \Gamma \) \( \leftarrow \) Select LST randomly \( / \) 2-Opt, Relocate, Swap \( [24], [29] - [31] \)
10: 10: \( R'(K, \mathcal{V}, \Psi, \Theta, \Gamma) \) \( \leftarrow \) Execute LST and update the string indices
11: 11: \( \Delta' \) \( \leftarrow \) Compute new time cost for \( \{ R'(K, \mathcal{V}, \Psi, \Theta, \Gamma) \} \)
12: 12: if \( \Delta' < \Delta \) then
13: 13: \( R(K, \mathcal{V}, \Psi, \Theta) \) \( \leftarrow \) \( R'(K, \mathcal{V}, \Psi, \Theta, \Gamma) \)
14: 14: \( \Delta \) \( \leftarrow \) \( \Delta' \)
15: 15: end if
16: 16: Repeat 5-15 with \( I_{RD} \) times
17: return \( R, \Delta \);
A. Preliminaries

1) Multi-cell setting: Fig. 3 illustrates a graph coloring of the typical multi-cell network model represented by (a) the SC model and (b) EC model, respectively. Observe that unlike the SC model, the cell coverage of the EC model generates overlapping areas, which is adjustable to be extended. Both models are illustrated as the graph coloring. In particular the EC model is illustrated as an order-3 Venn diagram with 7 coloured regions; creating in essence virtual cells. To this end, as observed in Fig. 3(b), we define \( \zeta_B = \{1, 2, ..., b\} \) as the set of certain CPs between two adjacent cells in the three different overlapping areas, i.e. the incidence vertices in the 2-colourable graph area, and we also denote \( \zeta_Q = \{1, 2, ..., q\} \) as the set of certain CPs among three cells in the overlapping area of three cells, i.e. the incidence vertices of the 3-colourable graph area. Which CPs in \( \zeta_B \) and \( \zeta_Q \) are allocated to a certain cell coverage, i.e. \( \psi_{\xi} \), results in transition of FBSs trajectory in each cell since CPs in \( \zeta_B \) and \( \zeta_Q \) in the EC model connect more potential feasible links that FBSs are able to use than them in the SC model. Therefore, this aspect significantly affects the performance of the different heuristic algorithms as detailed in the sequel.

2) Division into two tiers: The proposed heuristic algorithms are in essence composed by two different tiers; the aerial plane and the terrestrial plane as shown in Fig. 1 in order to increase decision making quality: 1) The first tier relates to the terrestrial plane: A method of clustering (selecting) CPs and a macro-BS to allocate \( \zeta_B \) and \( \zeta_Q \) to each given cell coverage area, which is in essence similar to a node clustering problem [32]. 2) The second tier relates to the aerial plane: A method for local search that creates FBS routes (paths).

B. Set of Heuristic Algorithms

With the above methods amalgamated in order to create feasible solutions, we present three flavours of heuristic algorithms as follows.

1) Randomized distribution with randomized local search (RD-RLS): Alg. 1 provides the pseudo code of RD-RLS heuristic algorithm and Fig. 4 shows a toy example with the FBSs trajectory represented by a set of string indices used in the proposed heuristic algorithms.

In the first tier, CPs and macro-BSs are randomly distributed in the multi-cell. Then, a group of both several CPs and a macro-BS are generated for each cell. Following that, each group is divided into the number of FBSs as a part of the group and each part is allocated to a string (route) index in a table, which is defined as \( R(k) \) (e.g. if \( K \in K = 3 \) is the number of string indices are three) for the FBSs trajectory. The set of string indices are composed of numbers for the FBS’s route in a one-dimensional vector. \( R(k) \) represents an FBSs trajectory such that for example, \( R(3) = \{[82 1 4 8]_1, [9 5 6 9]_2, [10 3 7 10]_3\} \), where the number 8, 9 and 10 denote macro-BSs (depots), and \( \cdot_{ik} \) is the string index for the FBS’s route and \( \cdot \) denotes CPs visited by each FBS in sequence. To this end, FBSs’ routes are initialized for the first time as referred in Fig. 7(a).

Algorithm 2 Brute Force with Randomized Local Search (BF-RLS)

1: \( \text{alg} \) ← Initialize iteration number, parameters in Table 1
2: First tier:
3: Generate \( \Psi, \kappa, \nu \)
4: \( B(\zeta_B, \zeta_Q) \) ← List all possible combinations for \( \zeta \) by using Brute Force search // The number of all cases is \( 3^2 B \)
5: \( R(k) \) ← Create a set of string indices in \( K \) rows
6: \( R(k, \Psi, \kappa, \nu) \) ← Insert CPs and macro-BSs that belong to each \( \psi \) except for the CPs in \( \zeta_B \) and \( \zeta_Q \) into the string indices
7: \( R(k, \Psi, \kappa, \nu) \) ← Combine \( R(k, \Psi, \kappa) \) with each case of \( B(\zeta_B, \zeta_Q) \)
8: Second tier:
9: \( R(k, \Psi, \kappa, \nu) \) ← Generate UAVs trajectories
10: \( \Delta ← \) Compute time cost for \( R(k, \Psi, \kappa, \nu) \)
11: \( \psi ← \) Select LST randomly // 2-Opt, Relocate, Swap [24, 29, 31]
12: \( \Delta ← \) Compute new time cost for \( R'(k, \Psi, \kappa, \nu) \)
13: if \( \Delta' < \Delta \) then
14: \( R(k, \Psi, \kappa, \nu) \) ← \( R'(k, \Psi, \kappa, \nu) \)
15: \( \Delta ← \Delta' \)
16: end if
17: Repeat 9-17 with \( I_{RD} \) times
18: \( \eta ← \min(\Delta) // \) Find a minimum time cost of the set of string indices
19: return \( R, \eta \);

Algorithm 3 Brute Force with Nearest Neighbour (BF-NN)

1: \( \Theta ← \) Initialize parameters in Table 1
2: First tier:
3: Generate \( \Psi, \kappa, \nu \)
4: \( B(\zeta_B, \zeta_Q) \) ← List all possible combinations for \( \zeta \) by using Brute Force search // The number of all cases is \( 3^2 B \)
5: \( R(k) \) ← Create a set of string indices in \( K \) rows
6: \( R(k, \Psi, \kappa) \) ← Insert CPs and macro-BSs that belong to each \( \psi \) except for the CPs in \( \zeta_B \) and \( \zeta_Q \) into the string indices
7: \( R(k, \Psi, \kappa) \) ← Combine \( R(k, \Psi, \kappa) \) with each case of \( B(\zeta_B, \zeta_Q) \)
8: Second tier:
9: \( R(k, \Psi, \kappa) \) ← Generate UAVs trajectories
10: \( R(k, \Psi, \kappa) \) ← Execute NN algorithm [33] and update the set of string indices
11: \( \Delta ← \) Compute time cost for \( R(k, \Psi, \kappa) \)
12: \( \eta ← \min(\Delta) // \) Find a minimum time cost of the set of string indices
13: return \( R, \eta \);

In the second tier, the set of the string indices is switched by \( LST \) \( \rightarrow \) 2-Opt, Relocate, Swap algorithms [24], [29]–[31] as illustrated in Fig. 8 as referred in Fig. 7(b) and the whole process is repeated iteratively denoted by the iteration number \( I_{RD} \). Finally, the best feasible trajectory planning is selected.

2) Brute Force with Randomized Local Search (BF-RLS): Alg. 2 provides the pseudo code of BF-RLS heuristic algorithm.

In the first tier, CPs and a macro-BS are randomly distributed in the multi-cell. Then, by using Brute-Force (BF) search, this algorithm lists all possible combinations for the CPs in the sets \( \zeta_B \) and \( \zeta_Q \). Following that, this algorithm allocates the CPs in the listed combinations into respective string indices of a table for the FBSs trajectory (i.e. the set of the string indices consists of \( K \) row vectors). In the next step, the rest of the CPs out of \( \zeta_B \) and \( \zeta_Q \) and the macro-BSs are sequentially distributed into the set of the string indices as shown in Fig. 7(b).

In the second tier, the set of the string indices is switched by the LST as shown in Fig. 7(b) and the process of the second
Table II represents the parameters used for performance evaluation. The number of FBSs \(K\) and the limited number of deployed FBSs \(\delta_i\) are fixed to one respectively in each cell, i.e. \(K = 3\), whilst the total number of cells (macro-BSs \(M\)) are assumed to be three to encapsulate a typical multi-cell set-up.\(^6\)

To focus on cell-edge CPs for all schemes except for OUT-MNB, we define \(\psi_{\text{sub}}\) as an area illustrated with the white hexagon shape in the center of the cell as shown in Fig. 9(b). The size of the length of \(\psi_{\text{sub}}\) is fixed to 240m, i.e. \(0.6L_H\), when \(L_H = 400\) as a baseline. Then, we also define \(\psi_{\text{areas}} = \psi_m - \psi_{\text{sub}}\), \(\psi_m \in \Psi\) illustrated with the grey colour in Fig. 9(b). This deployment area \(\psi_{\text{areas}}\) is applied to all schemes. We note that the grey colour is not illustrated in the other different figures in order to provide clear figures. Furthermore, to compete with FBSs for service, the CPs are randomly distributed within the region of a third in hexagon cells, i.e. adjacent areas between cells, as illustrated with magenta, blue and cyan dotted line in Fig.9(b) to 9(g). Thus, the CPs in those regions are served by FBSs potentially located at different cells. In addition, and without loss of generality, we assume that all macro-BSs are located at the center of each cell and the iteration number \(I_{\text{RD}}\) and \(I_{\text{BF}}\) defined for RD-RLS and BF-RLS algorithms are 10000 and 1000 respectively. To intuitively observe the performance of our design, we assume that all GUs at each CP have identical average throughput requirements, i.e. \(Q_i = Q_{f,\text{sub}}\). Then, \(Q_i\) is fixed to 25 Mbits. Moreover, we assume, without loss of generality, that the time window for the QoS support for all CPs is identical, i.e. \(e_i = e\) and \(l_i = l\).

We vary the side length of the cells \(L_H\) within the range of 400m to 600m and \(L_H = 400\) is standardized as the SC model for OUT-MSC. Whereas, \(L_H > 400\) are considered as the EC model for OUT-MEC. We note that all heuristic algorithms are evaluated with \(L_H = 480\). Furthermore, we set OUT-MNB and OUT-MSC schemes as benchmarks in terms of performance since their solutions do not take into account FBS deployment in the cell overlapping coverage areas; even though they do take into account the multi-cell network model. Henceforth, OUT-MNB and OUT-MSC can be regarded as the benchmarks compared to our proposed schemes, i.e. the OUT-MEC and the proposed heuristic algorithms.

V. Numerical Investigations

In this section, a Monte Carlo simulation with 500 events is performed to evaluate, validate and analyze the effectiveness of the proposed optimal and sub-optimal (heuristics) trajectory optimization schemes in a nominal multi-cell wireless network through a number of different use case\(^7\).

We adopt a model of a UAV with rotary wings\(^25\). In\(^25\), the UAV with rotary wings is deployed with two portions of energy consumption, i.e. traveling energy and hovering energy. Moreover, the authors\(^25\) derived the optimal velocity of the UAV that minimizes the energy consumption while maximizing the UAV endurance. This modeling can be applied to our system as well. Therefore, we assume that the velocity \(v\) of the FBS is the optimal value, i.e. 10.21 m/s (further details can be seen in\(^25\) Fig. 2)).

\(^6\)The simulation operated on MATLAB is stopped at 120 minutes running time, which is the case when the following parameters were combined and exceeded these values: \(N > 18\), \(K > 3\).
the OUT-MSC scheme for the case of \( \omega \) achieves the best performance by gaining up to 31.2% and under investigation. Recognize that the OUT-MNB scheme the different number of CPs \( N \) the FBSs have managed to collect data from all CPs per \( \omega \) mission, i.e. all FBSs completing their Hamiltonian paths. To this end, we define a result of the optimization solution.

Fig. 9. Examples of FBSs trajectories for all different schemes in the case of \( K = 3, N = 9, \) and \( M = 3 \).

The OUT-MNB visits the majority of CPs as shown in Fig. 9(a). It is noticeable that only one FBS in the OUT-MNB visits the majority of CPs as shown in Fig. 9(a).

In other words, the OUT-MNB scheme achieves an overly unfair trajectory, i.e. the FBS trajectory departed from the number 11 (macro-BS). This is because the FBSs are able to fly over the whole area without any boundaries of cells as a result of the optimization solution.

We evaluate the effectiveness of the proposed schemes in the multi-cell network setting. To this end, we define \( \omega_{\text{off}} = \sum_{i=1}^{N} Q_i / \text{TTT} \) [bps], which measures how efficiently the FBSs have managed to collect data from all CPs per mission, i.e. all FBSs completing their Hamiltonian paths. Hence, larger values of \( \omega_{\text{off}} \) reflect a more efficient allocation of the FBSs.

The OUT-MNB scheme [120%] shows that the TTT and \( \omega_{\text{off}} \) substantially gain approximately 23.1% and 19.2% respectively for the case of \( N = 9 \) compared with the OUT-MSC scheme. It is also worth pointing out that both the TTT and \( \omega_{\text{off}} \) metric of the BF-NN attain significant performance by improving up to 22% and 18.1% gain respectively compared with the OUT-MSC in the entire simulations, which is substantially compatible with the OUT-MEC [120%]. Whereas, the TTT and \( \omega_{\text{off}} \) of both BF-

Hereafter, '[\%]' is attached to the OUT-MEC solution according to the size of \( L_M \), e.g. OUT-MEC [110%] for the case of \( L_M = 440 \).
RLS and RD-RLS are greater than the OUT-MEC [110%] for the case of \( N = 9 \) and equivalent to the OUT-MEC [110%] for the case when \( N = 12 \) up to 15 although the performance slightly deteriorates when \( N = 18 \).

Table [11] presents the upper and lower bound values of the \( \omega_{off} \) metric for all schemes. Observe that in terms of the lower bound of the \( \omega_{off} \), all proposed heuristic algorithms outperform the baseline, OUT-MSC, and their performance lie in between OUT-MEC [110%] and OUT-MEC [120%] over the entire cases for \( N = 9 \) to 18. Whereas, in terms of the upper bound, the RD-RLS and BF-RLS gain worse performance for the case of \( N = 12 \) to 18 than the OUT-MSC. Yet, the BF-NN gains better performance than the OUT-MSC in the entire set of simulations and compatible with the OUT-MNB and OUT-MEC [120%] schemes.

B. Fairness and Mission Completion Time

We investigate how fair and balanced each FBS’s trajectory is (i.e. in terms of TTT of each FBS) for the case of \( K = 3 \) and using the Jain’s fairness index with results shown in Fig. 12. The Jain’s fairness index [34] is a metric widely used to measure fairness for the different solutions; it has a range between \([0, 1]\) and the larger the fairness index is, the fairer the allocation is; in this case the FBSs trajectory in terms of TTT. All the proposed schemes obtain more than 0.71 value over the entire set of simulations. The OUT-MSC obtains the highest and fairest Jain index gaining around 1. All heuristic schemes have performance compatible between the OUT-MEC [110%] and the OUT-MEC [120%] by setting their \( L_H \) value as 480m. However, the OUT-MNB achieves below 0.52 that is the lowest value compared with other different schemes throughout the whole simulations since there are no boundary restrictions for the FBSs deployment and it shows a decrease tendency along with the increase on the number of CPs.

Fig. 13 represents mission completion time of FBSs, which is defined as a mission time until all FBSs return to the macro-BS; an important metric for the efficient operation of the network which has been widely used as discussed in [12]. As can be seen from Fig. 13 the OUT-MSC achieves the lowest mission completion time by gaining 83.2s at \( N = 9 \) to 103.9s at \( N = 18 \) whilst the OUT-MEC [110%] achieves similar results across the set of conducted simulations. It is noticeable that the OUT-MEC [120%] and the BF-NN achieves a similar performance to the OUT-MSC, and attains similar values, i.e. 121.9s and 120.5s respectively, at \( N = 18 \). Moreover, the RD-RLS also attains comparable values to the OUT-MSC at \( N = 9 \) to 12, although the gap between them is slightly wider from \( N = 15 \) onward. On the other hand, the OUT-MNB and the BF-RLS suffer a steep increase in the mission completion time, which mounts up to 171.2s and 151.2s respectively for the case of \( N = 18 \).

C. Extension to Coverage

In addition to the mission completion time, it is worth delving into the extension to coverage aspects. To this end, Fig. 14, Fig. 15 and 16 compare Jain’s fairness index, TTT and \( \omega_{off} \) with respect to varying the \( L_H \) from 400m to 600m (i.e. OUT-MSC to OUT-MEC [150%]) according to the number of CPs. In Fig. 15, observe that as the \( L_H \) is increased, the index value is decreased. Also, the values of the OUT-MEC [130%] to the OUT-MEC [150%] drops below 0.68 in the whole set of simulations and across all number of CPs. Accordingly, as the range of the coverage is extended and the overlapping region between cells is increased, the observed unfair trajectory of FBSs on TTT is generated.

As observed in Fig. 15 the TTT shows a decreasing trend with increasing \( L_H \), i.e. OUT-MSC to OUT-MEC [150%]. On the other hand, in Fig. 16 \( \omega_{off} \) represents a steadily increasing trend for higher values of \( L_H \).

D. Scalability of the Optimization Problems and Underlying Complexity Analysis

To provide a holistic view on the different schemes, we compare in this section the computational complexity and the scalability of the different schemes including the ones providing optimal solutions for low to medium network instances where the integer mathematical programs could be solved to optimality. To start with, the dimensionality of the optimization problems OUT-MNB and OUT-MSC/MEC, which define their scalability, is as follows,

- OUT-MNB: \(|V_c| + |V_d| + 2|V| + |K||V| + 2|K||V_c||V_d| + |K||V_c||V_d||F| + 2(|V_c|^2)|F| + |K||(V^2)|

Fig. 11. \( \omega_{off} \) versus the number of CPs (\( N \)) for all different schemes.

Fig. 12. Jain’s fairness index on TTT for all approaches.
Mission completion time (s)

<table>
<thead>
<tr>
<th></th>
<th>OUT-MSC</th>
<th>OUT-MEC[110%]</th>
<th>OUT-MEC[120%]</th>
<th>OUT-MEC[130%]</th>
<th>OUT-MEC[140%]</th>
<th>OUT-MEC[150%]</th>
<th>OUT-MNB</th>
<th>RD-RLS</th>
<th>BF-RLS</th>
<th>BF-NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUT-MSC</td>
<td>0.55</td>
<td>0.65</td>
<td>0.75</td>
<td>10.5</td>
<td>0.85</td>
<td>160</td>
<td>OUT-MSC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT-MEC[110%]</td>
<td>200</td>
<td>110</td>
<td>120</td>
<td>220</td>
<td>130</td>
<td>240</td>
<td>OUT-MEC[110%]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT-MEC[120%]</td>
<td>150</td>
<td>150</td>
<td>260</td>
<td>160</td>
<td>280</td>
<td>170</td>
<td>OUT-MEC[120%]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT-MEC[130%]</td>
<td>140</td>
<td>140</td>
<td>180</td>
<td>300</td>
<td></td>
<td></td>
<td>OUT-MEC[130%]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT-MEC[140%]</td>
<td>130</td>
<td>130</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OUT-MEC[140%]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT-MEC[150%]</td>
<td>120</td>
<td>120</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OUT-MEC[150%]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Further, in terms of the underlying complexity using the big-O notation, the proposed RD-RLS, BF-RLS and BF-NN have a limiting behaviour of $O(I_{RD}(N + M)^2)$, $O(I_{BF}3^{q/2}b(N + M)^2)$ and $O(3^{q/2}b(N + M)^2)$, respectively, which are simply computed by parameters $I_{RD}$, $I_{BF}$, $q$, $b$, $N$ and $M$. Specifically, $I_{RD}$ and $I_{BF}$ of the complexity represent the iteration number and $3^{q/2}b$ represents the number of all combination cases of $\zeta_Q$ and $\zeta_B$ in the heuristic algorithms. Also, $(N + M)^2$ represents the worst case complexity of the local search techniques (LST), i.e. 2-Opt [35], and the Nearest Neighbor (NN) [36]. We note that the Brute Force (BF) search method has an exponentially increased computational complexity in general such as $O(3^{q/2}b)$ and this is applicable to both BF-RLS and BF-NN algorithms. However, since $q$ and $b$ represent only the CPs in $\zeta_B$ and $\zeta_Q$, they are a relatively small number. In general, since the OUT-MNB and the OUT-MSC/MEC problems are NP-hard (the detailed proof can be seen in Appendix A), they require significant higher computational time than the heuristic schemes with the same parameters as shown in Fig. [7]. Moreover, and as expected, the computation time for all schemes shows a growing trend as the number of problem parameters is increased. The OUT-MNB scheme requires approximately 1240s computational time (on average) when $N = 18$, which is the upper bound for all schemes. On the other hand, all proposed heuristic algorithms provide competitive solutions in low computation time even for relatively larger network instances and they require only a few seconds to provide decision making in all the scenarios. In particular, the BF-NN requires less than only 0.18s throughout the entire simulations and much lower than the OUT-MNB.
by improving nearly 6889 times decrease when $N = 18$. Therefore, even though these heuristic algorithms suffer a slight loss in terms of network capacity, i.e. TTT and fairness of FBS trajectory, their performance is competitive, whilst they allow real-time decision making. Hence, the proposed schemes can be consided as part of UAV traffic management functionalities which are expected to be integrated within network orchestration functionalities, e.g. under a nominal Mobile Edge Computing (MEC) or Network Function Virtualization (NFV) environment.

VI. CONCLUSIONS

This paper presents a novel optimization framework to compute optimal path planning trajectories for multiple UAVs operating as flying base stations (FBSs) serving ground users in a multi-cell network layout where macro-BSs act as depots for the UAVs. The key novelty of the proposed framework is that it allows for the creation of virtual cell boundaries so that the actual energy consumption of the FBSs. Furthermore, the proposed heuristic algorithms can be deemed as being scalable and amenable for real-time implementation due to the low underlying computational complexity cost.

The outcome of this work motivates future works in UAV-aided 5G networks. One promising problem is that multiple FBSs return to a close depot not returning to the original depot once the FBSs serve GUUs, which requires adaptive optimization techniques but might further improve the UAV’s performance in terms of deployment.

APPENDIX

A. Proof of NP-hardness

Lemma 2. The OUT-MNB and OUT-MSC/MEC schemes are both NP-hard problems.

Proof. Both the OUT-MNB and the OUT-MSC/MEC optimization problems include the constraints in $\text{(4a)}$ and these constraints reflect the FBS trajectory route construction. Whereas, the constraints in $\text{(5c)}$ relate to the time window to ensure that a Hamiltonian cycle (path) is created via sub-tour elimination [27] by using Miller-Tucker-Zemlin (MTZ) formulation [28]. By eliminating the time window constraints, the optimization problems reduce to the VRP, which is a subset of the TSP, since their objective function seeks for paths (via the decision variable $x$) with reduced aggregate total travel time.

Hence, the OUT-MNB and OUT-MSC/MEC optimization problems in the form of a MILP encapsulate a TSP-like tour in terms of trajectory planning. As a result, they are as hard as the TSP problem and therefore fall within the well-known NP-hard [10] set of problems.

REFERENCES
